

# CAMPUS LIFE

Trabajo Fin de Grado



Universidad  
Carlos III de Madrid

Nombre: Daniel Sánchez Muñoz  
Titulación: Ingeniería Informática  
Tutor: Javier Fernández Muñoz



**TÍTULO:** Diseño e implementación de Campus Life, una aplicación móvil para la consulta e inscripción de actividades deportivas y culturales.

**AUTOR:** SÁNCHEZ MUÑOZ, DANIEL.

**DIRECTOR:** FERNANDÉZ MUÑOZ, JAVIER.

## EL TRIBUNAL

**PRESIDENTE:** CALLE GOMEZ, FRANCISCO JAVIER.

**SECRETARIO:** GARCIA BLAS, FRANCISCO JAVIER.

**VOCAL:** CRUZ FERNANDEZ, ROSA MARIA DE LA.

Realizado el acto de defensa y lectura del Trabajo Fin de Grado el día 4 de Octubre de 2016 en Leganés (Madrid), en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la calificación de:

PRESIDENTE:

SECRETARIO:

VOCAL:



## AGRADECIMIENTOS.

La creación de esta aplicación, es un conjunto del trabajo realizado por numerosas personas y diferentes departamentos, que han estado unidos durante las diversas fases del análisis, diseño e implementación.

Me gustaría realizar una mención especial sobre mis compañeros de departamento con los que he trabajado mano a mano en este proyecto y sin los cuales no habría salido adelante.

En primera instancia, gracias a Enrique Bueno por darnos la oportunidad de embarcarnos en el mundo de las aplicaciones móviles y aprender de los mejores, sobre las nuevas tecnologías.

Gracias a David Fernández por confiar en nosotros y darnos la oportunidad de poder aprender de sus conocimientos.

Gracias a Jorge Hermosilla por acompañarme en este proyecto de principio a fin, realizando en paralelo el desarrollo de la app para Android y apoyándome cuando las cosas no salían adelante.

Me gustaría dar también las gracias a Javier Fernández por ser mi tutor en este proyecto y confiar en mí cuando otros no me dieron la oportunidad.

Por otro lado me gustaría dar las gracias a mi mujer Sonia por apoyarme en todo momento, por animarme a seguir hacia adelante y por estar pendiente de mí cuando no rendía a mi máximo nivel, gracias por aguantar tanto y por convertir mis problemas en tus prioridades.

Por último gracias a mis padres, por apoyarme en todas mis decisiones, gracias por mirar siempre por mí, aunque eso supusiera tenerme lejos, gracias por ayudarme a cambiar de vida y a inculcarme los valores positivos que poseo.



## RESUMEN.

El siguiente proyecto explica el desarrollo de Campus Life, una aplicación compatible con la plataforma móvil de Apple (iOS), diseñada para dar soluciones a las necesidades del departamento Espacio Estudiantes (ESES) perteneciente a la Universidad Carlos III de Madrid (UC3M). Permite realizar consultas e inscripciones sobre actividades deportivas, culturales y de orientación, organizadas por ESES.

Campus Life es una aplicación gratuita y de ámbito educativo pero no es accesible para todos los usuarios, la aplicación tiene acceso restringido a estudiantes, profesores y personal de administración y servicios de la UC3M.

Sus principales funcionalidades y servicios son:

- Solicitar tu Tarjeta de Espacio Estudiantes (TE), que permite obtener descuentos en la Universidad y de las ventajas del carnet ISIC-ITIC.
- Consultar promociones.
- Informarse de la oferta de actividades, las cuales están organizadas por campus y tipos de actividad. De cada actividad se puede consultar una breve descripción con una imagen, el horario y el precio.
- Inscribirse en las actividades.
- Marcar actividades como favoritas para poder inscribirte a posteriori o compartir por redes sociales.
- Acceder a tus actividades inscritas.
- Contactar con el departamento de ESES así como geo-localizar sus emplazamientos en los diferentes campus y acceder a sus perfiles en las diferentes redes sociales.

El presente documento tiene como finalidad la presentación del proceso completo llevado a cabo para la realización del proyecto.





# ÍNDICE DE CONTENIDOS.

<b>AGRADECIMIENTOS.....</b>	<b>5</b>
<b>RESUMEN.....</b>	<b>7</b>
<b>ÍNDICE DE ILUSTRACIONES.....</b>	<b>12</b>
<b>ÍNDICE DE TABLAS .....</b>	<b>14</b>
<b>CAPÍTULO 1: INTRODUCCIÓN Y OBJETIVOS. ....</b>	<b>19</b>
1.1 VISIÓN GENERAL.....	20
1.2 MOTIVACIÓN.....	20
1.3 OBJETIVOS.....	21
1.4 FASES DEL DESARROLLO.....	22
1.5 ESTRUCTURA DEL DOCUMENTO.....	23
1.5.1 CAPÍTULO 1, Introducción y objetivos.....	23
1.5.2 CAPÍTULO 2, Estado de la cuestión.....	23
1.5.3 CAPÍTULO 3, Análisis.....	23
1.5.4 CAPÍTULO 4, Diseño.....	23
1.5.5 CAPÍTULO 5, Implementación y Pruebas.....	24
1.5.6 CAPÍTULO 6, Planificación y presupuesto.....	24
1.5.7 CAPÍTULO 7, Conclusiones y próximas novedades.....	24
<b>CAPÍTULO 2. ESTADO DE LA CUESTIÓN.....</b>	<b>26</b>
2.1 EVOLUCIÓN DE LA TECNOLOGÍA.....	27
2.1.1 ORIGEN.....	27
2.1.2 INTERNET.....	28
2.1.3 WEB 2.0.....	28
2.1.4 MÓVILES Y SMARTPHONES.....	28
2.2 ANÁLISIS DE LA PLATAFORMA MÓVIL.....	29
2.2.1 ANDROID.....	30
2.2.2 iOS.....	31
2.2.3 WINDOWS PHONE.....	32
2.2.4 BLACKBERRY.....	33
2.2.5 SYMBIAN.....	34
2.2.6 FIREFOX OS.....	35
2.2.7 UBUNTU TOUCH.....	36
2.3 CRITERIOS DE SELECCIÓN DEL SISTEMA OPERATIVO.....	39
2.3.1 PRIMERA NECESIDAD.....	39
2.3.2 SEGUNDA NECESIDAD.....	39
2.3.3 TERCERA NECESIDAD.....	39
2.4 CONCLUSIÓN SOBRE LA ELECCIÓN DEL SISTEMA OPERATIVO.....	40
2.4.1 PRIMERA CONCLUSIÓN.....	40
2.4.2 SEGUNDA CONCLUSIÓN.....	41
2.4.3 TERCERA CONCLUSIÓN.....	41
2.4.4 CUARTA CONCLUSIÓN.....	42
2.4.5 ELECCIÓN DEFINITIVA, iOS.....	43
2.5 ANÁLISIS DE APLICACIONES.....	45
2.6 TECNOLOGÍA UTILIZADA.....	49

2.6.1 HERRAMIENTAS BÁSICAS.....	50
2.6.2 ENTORNO DE DESARROLLO: XCode.....	50
2.6.3 LENGUAJE DE DESARROLLO: OBJECTIVE-C.....	51
2.6.4 COCOA PODS.....	52
2.6.5 WEB SERVICES.....	54
2.6.6 INTELLIJ IDEA.....	55
2.6.7 SPRING BOOT.....	56
2.6.8 SERVIDOR WEB.....	56
2.6.9 GESTOR DE BASE DE DATOS.....	57
<b>CAPÍTULO 3: ANÁLISIS DEL SISTEMA.....</b>	<b>59</b>
3.1 REQUISITOS DE USUARIO.....	60
3.1.1 REQUISITOS DE CAPACIDAD.....	61
3.1.2 REQUISITOS RESTRICCIÓN.....	70
3.2 CASOS DE USO.....	73
3.3 REQUISITOS DE SISTEMA.....	88
3.3.1 REQUISITOS FUNCIONALES.....	89
3.3.2 REQUISITOS NO FUNCIONALES.....	100
3.4 MATRIZ DE TRAZABILIDAD.....	110
<b>CAPÍTULO 4: DISEÑO DEL SISTEMA.....</b>	<b>117</b>
4.1 DISEÑO DE LA ARQUITECTURA.....	118
4.1.1 PROCESO DE PETICIONES.....	118
4.1.2 PATRÓN MVC.....	119
4.1.3 INTEGRACIÓN MVC – CLIENTE SERVIDOR.....	120
4.1.4 LISTA DE PETICIONES.....	121
4.1.5 COMPONENTES DE LA ARQUITECTURA.....	123
4.1.6 MATRIZ DE TRAZABILIDAD.....	127
4.2 DISEÑO DE LA BASE DE DATOS.....	128
4.2.1 ENTIDAD ACTIVIDAD.....	129
4.2.2 ENTIDAD USUARIO.....	130
4.2.3 ENTIDAD INSCRIPCIÓN.....	130
4.2.4 ENTIDAD SECCIÓN.....	131
4.2.5 ENTIDAD HORARIO.....	131
4.2.6 ENTIDAD PROMOCIÓN.....	132
4.2.7 ENTIDAD COMPRA TE.....	132
4.2.8 ENTIDAD TARJETAS.....	133
4.3 DISEÑO DE LA INTERFAZ.....	134
4.3.1 DISEÑO DE LAS VISTAS.....	135
4.4 DIAGRAMA DE NAVEGACIÓN.....	153
<b>CAPÍTULO 5: IMPLEMENTACIÓN Y PRUEBAS.....</b>	<b>156</b>
5.1. IMPLEMENTACIÓN.....	157
5.2 PRUEBAS.....	162
5.2.1 PRUEBAS FUNCIONALES.....	163
5.2.2 MATRIZ DE TRAZABILIDAD.....	182
5.2.3 PRUEBAS DE ESTRÉS.....	184
<b>CAPÍTULO 6: PLANIFICACIÓN Y PRESUPUESTO.....</b>	<b>187</b>
6.1 PLANIFICACIÓN.....	188
6.1.1 DIAGRAMA DE GANTT.....	191
6.2 PRESUPUESTO.....	192
6.2.1 TIEMPO DE FASES.....	192
6.2.2 COSTE DEL PERSONAL.....	193
6.2.3 COSTE DEL HARDWARE.....	195

6.2.4 COSTE DEL SOFTWARE Y LICENCIAS.....	196
6.2.5 COSTE TOTAL DEL PROYECTO.....	196
<b>CAPÍTULO 7: CONCLUSIÓN Y PRÓXIMAS NOVEDADES. ....</b>	<b>199</b>
7.1 CONCLUSIONES.....	200
7.1.1 PROYECTO.....	200
7.1.2 PERSONALES.....	201
7.2 PRÓXIMAS NOVEDADES.....	202
<b>GLOSARIO. ....</b>	<b>205</b>
<b>BIBLIOGRAFÍA. ....</b>	<b>210</b>
<b>ABSTRACT. ....</b>	<b>213</b>
<b>INTRODUCTION AND OBJECTIVES. ....</b>	<b>214</b>
1.1 OVERVIEW.....	214
1.2 MOTIVATION.....	214
1.3 OBJETIVES.....	215
1.4 DEVELOPMENT STAGES. ....	216
1.5 DOCUMENT STRUCTHURE.....	217
1.5.1 CHAPTER 1, Introduction and objectives.....	217
1.5.2 CHAPTER 2, State of affairs.....	217
1.5.3 CHAPTER 3, Analysis. ....	217
1.5.4 CHAPTER 4, Design. ....	217
1.5.5 CHAPTER 5, Implementation and testing.....	218
1.5.6 CHAPTER 6, Planning and budgeting.....	218
1.5.7 CHAPTER 7, Conclusions and coming soon.....	218
<b>CONCLUSION AND COMING SOON.....</b>	<b>219</b>
2.1 CONCLUSIONS. ....	219
2.1.1 PROJECT. ....	219
2.1.2 PERSONAL.....	220
2.2 COMING SOON. ....	221



## ÍNDICE DE ILUSTRACIONES

Ilustración 1: iOS 9.....	31
Ilustración 2: Windows Phone.....	32
Ilustración 3: BlackBerry OS. ....	33
Ilustración 4: Symbian OS. ....	34
Ilustración 5: FireFox OS.....	35
Ilustración 6: Ubuntu Touch.....	36
Ilustración 7: Cuota de mercado de los smarphones en diferentes regiones.....	37
Ilustración 8: Características de los principales Sistemas Operativos del mercado. ....	38
Ilustración 9: Especificaciones de los dispositivos elegidos para el desarrollo.....	44
Ilustración 10: Universidad Complutense de Madrid. ....	45
Ilustración 11: Universidad Autónoma de Madrid. ....	46
Ilustración 12: Universidad de Alcalá de Henares. ....	47
Ilustración 13: iPhone 6. ....	49
Ilustración 14: Objective-C vs Swift.....	51
Ilustración 15: Cocoa Pods.....	52
Ilustración 16: RESful.....	55
Ilustración 17: IDE IntelliJ IDEA.....	55
Ilustración 18: Spring Boot.....	56
Ilustración 19: Tomcat. ....	56
Ilustración 20: MySQL. ....	57
Ilustración 21: Cliente servidor.....	119
Ilustración 22: MVC.....	120
Ilustración 23: Arquitectura. ....	123
Ilustración 24: Modelo Relacional de la BBDD.....	129
Ilustración 25: Login.....	136
Ilustración 26: Actividades.....	137
Ilustración 27: Ventanas emergentes.....	139
Ilustración 28: Favoritas. ....	141
Ilustración 29: Mis Actividades 1. ....	143
Ilustración 30: Mis Actividades 2. ....	144
Ilustración 31: Promociones 1. ....	145
Ilustración 32: Promociones 2. ....	146
Ilustración 33: Contacto. ....	147
Ilustración 34: Mapas. ....	148
Ilustración 35: Información. ....	149
Ilustración 36: Inscripción. ....	150
Ilustración 37: Tarjetas. ....	151
Ilustración 38: Diagrama de navegación. ....	153
Ilustración 39: Prueba login.....	163
Ilustración 40: Prueba login 1. ....	164
Ilustración 41: Prueba login 2. ....	165
Ilustración 42: Prueba HUD y botón de recarga.....	166

Ilustración 43: Estrella favorito desmarcada.....	168
Ilustración 44: Estrella favorito marcada. ....	168
Ilustración 45: Prueba Favoritos. ....	169
Ilustración 46: Prueba descripción, nombre y precios. ....	170
Ilustración 47: Prueba tipos de actividad. ....	171
Ilustración 48: Prueba Horario. ....	172
Ilustración 49: Prueba Actividades.....	173
Ilustración 50: Prueba Menú.....	174
Ilustración 51: Prueba Favoritas. ....	176
Ilustración 52: Prueba Promociones.....	177
Ilustración 53: Prueba Contacto.....	178
Ilustración 54: Prueba Inscripción.....	179
Ilustración 55: Prueba Información.....	180
Ilustración 56: Prueba Compartir. ....	181

## ÍNDICE DE TABLAS

Tabla 1 Descripción Tabla de Requisitos de Usuario.....	60
Tabla 2: Requerimiento de Usuario Capacidad 1.....	61
Tabla 3: Requerimiento de Usuario Capacidad 2.....	61
Tabla 4: Requerimiento de Usuario Capacidad 3.....	62
Tabla 5: Requerimiento de Usuario Capacidad 4.....	62
Tabla 6: Requerimiento de Usuario Capacidad 5.....	62
Tabla 7: Requerimiento de Usuario Capacidad 6.....	63
Tabla 8: Requerimiento de Usuario Capacidad 7.....	63
Tabla 9: Requerimiento de Usuario Capacidad 8.....	64
Tabla 10: Requerimiento de Usuario Capacidad 9.....	64
Tabla 11: Requerimiento de Usuario Capacidad 10.....	64
Tabla 12: Requerimiento de Usuario Capacidad 11.....	64
Tabla 13: Requerimiento de Usuario Capacidad 12.....	65
Tabla 14: Requerimiento de Usuario Capacidad 12.....	65
Tabla 15: Requerimiento de Usuario Capacidad 14.....	65
Tabla 16: Requerimiento de Usuario Capacidad 15.....	66
Tabla 17: Requerimiento de Usuario Capacidad 16.....	66
Tabla 18: Requerimiento de Usuario Capacidad 17.....	67
Tabla 19: Requerimiento de Usuario Capacidad 18.....	67
Tabla 20: Requerimiento de Usuario Capacidad 19.....	68
Tabla 21: Requerimiento de Usuario Capacidad 20.....	68
Tabla 22: Requerimiento de Usuario Capacidad 21.....	68
Tabla 23: Requerimiento de Usuario Capacidad 22.....	69
Tabla 24: Requerimiento de Usuario Capacidad 23.....	69
Tabla 25: Requerimiento de Usuario Capacidad 24.....	69
Tabla 26: Requerimiento de Usuario Restricción 1.....	70
Tabla 27: Requerimiento de Usuario Restricción 2.....	70
Tabla 28: Requerimiento de Usuario Restricción 3.....	71
Tabla 29: Requerimiento de Usuario Restricción 4.....	71
Tabla 30: Requerimiento de Usuario Restricción 5.....	71
Tabla 31: Requerimiento de Usuario Restricción 6.....	71
Tabla 32: Requerimiento de Usuario Restricción 7.....	72
Tabla 33: Requerimiento de Usuario Restricción 8.....	72
Tabla 34 Descripción Tabla de Casos de Uso.....	73
Tabla 35: Caso de uso 1.....	76
Tabla 36: Caso de uso 2.....	76
Tabla 37: Caso de uso 3.....	76
Tabla 38: Caso de uso 4.....	77
Tabla 39: Caso de uso 5.....	77
Tabla 40: Caso de uso 6.....	78
Tabla 41: Caso de uso 7.....	78
Tabla 42: Caso de uso 8.....	78
Tabla 43: Caso de uso 9.....	79

Tabla 44: Caso de uso 10.....	79
Tabla 45: Caso de uso 11.....	79
Tabla 46: Caso de uso 12.....	80
Tabla 47: Caso de uso 13.....	80
Tabla 48: Caso de uso 14.....	80
Tabla 49: Caso de uso 15.....	81
Tabla 50: Caso de uso 16.....	81
Tabla 51: Caso de uso 17.....	81
Tabla 52: Caso de uso 18.....	82
Tabla 53: Caso de uso 19.....	82
Tabla 54: Caso de uso 20.....	83
Tabla 55: Caso de uso 21.....	83
Tabla 56: Caso de uso 22.....	83
Tabla 57: Caso de uso 23.....	84
Tabla 58: Caso de uso 24.....	84
Tabla 59: Caso de uso 25.....	85
Tabla 60: Caso de uso 26.....	85
Tabla 61: Caso de uso 27.....	85
Tabla 62: Caso de uso 28.....	86
Tabla 63: Caso de uso 29.....	86
Tabla 64: Caso de uso 30.....	86
Tabla 65: Caso de uso 31.....	87
Tabla 66: Caso de uso 32.....	87
Tabla 67: Descripción Tabla de Requisitos de Sistema.....	88
Tabla 68: Requerimiento de Sistema Funcional 1. ....	89
Tabla 69: Requerimiento de Sistema Funcional 2. ....	89
Tabla 70: Requerimiento de Sistema Funcional 3. ....	90
Tabla 71: Requerimiento de Sistema Funcional 4. ....	90
Tabla 72: Requerimiento de Sistema Funcional 5. ....	91
Tabla 73: Requerimiento de Sistema Funcional 6. ....	91
Tabla 74: Requerimiento de Sistema Funcional 7. ....	92
Tabla 75: Requerimiento de Sistema Funcional 8. ....	92
Tabla 76: Requerimiento de Sistema Funcional 9. ....	93
Tabla 77: Requerimiento de Sistema Funcional 10.....	94
Tabla 78: Requerimiento de Sistema Funcional 11.....	94
Tabla 79: Requerimiento de Sistema Funcional 12.....	95
Tabla 80: Requerimiento de Sistema Funcional 13.....	95
Tabla 81: Requerimiento de Sistema Funcional 14.....	96
Tabla 82: Requerimiento de Sistema Funcional 15.....	96
Tabla 83: Requerimiento de Sistema Funcional 16.....	97
Tabla 84: Requerimiento de Sistema Funcional 17.....	98
Tabla 85: Requerimiento de Sistema Funcional 18.....	98
Tabla 86: Requerimiento de Sistema Funcional 19.....	99
Tabla 87: Requerimiento de Sistema Funcional 20.....	99
Tabla 88: Requerimiento de Sistema No Funcional 1.....	100
Tabla 89: Requerimiento de Sistema No Funcional 2.....	101
Tabla 90: Requerimiento de Sistema No Funcional 3.....	102



Tabla 91: Requerimiento de Sistema No Funcional 4.....	102
Tabla 92: Requerimiento de Sistema No Funcional 5.....	103
Tabla 93: Requerimiento de Sistema No Funcional 6.....	104
Tabla 94: Requerimiento de Sistema No Funcional 7.....	104
Tabla 95: Requerimiento de Sistema No Funcional 8.....	105
Tabla 96: Requerimiento de Sistema No Funcional 9.....	105
Tabla 97: Requerimiento de Sistema No Funcional 10. ....	106
Tabla 98: Requerimiento de Sistema No Funcional 11. ....	106
Tabla 99: Requerimiento de Sistema No Funcional 12. ....	107
Tabla 100: Requerimiento de Sistema No Funcional 13.....	108
Tabla 101: Requerimiento de Sistema No Funcional 14.....	108
Tabla 102: Requerimiento de Sistema No Funcional 15.....	109
Tabla 103: Requerimiento de Sistema No Funcional 16.....	109
Tabla 104: Requerimiento de Sistema No Funcional 17.....	109
Tabla 105: Matriz RUC – RSF.....	112
Tabla 106: Matriz RUC – RSNF.....	114
Tabla 107: Matriz RUR – RSF/RSNF .....	115
Tabla 108: Descripción Tabla de Componentes.....	123
Tabla 109: Componente de Actividad.....	124
Tabla 110: Componente de Promociones.....	125
Tabla 111: Componente de Tarjetas.....	125
Tabla 112: Componente de Inscripción.....	125
Tabla 113: Componente de Usuario.....	126
Tabla 114: Componente de Contacto.....	126
Tabla 115: Componente de Menú.....	126
Tabla 116: Matriz COM - RSF.....	127
Tabla 117: Descripción Tabla de Pruebas funcionales.....	162
Tabla 118: Prueba Funcional 1.....	164
Tabla 119: Prueba Funcional 2.....	165
Tabla 120: Prueba Funcional 3.....	166
Tabla 121: Prueba Funcional 4.....	167
Tabla 122: Prueba Funcional 5.....	168
Tabla 123: Prueba Funcional 6.....	169
Tabla 124: Prueba Funcional 7.....	170
Tabla 125: Prueba Funcional 8.....	171
Tabla 126: Prueba Funcional 9.....	172
Tabla 127: Prueba Funcional 10.....	173
Tabla 128: Prueba Funcional 11.....	174
Tabla 129: Prueba Funcional 12.....	175
Tabla 130: Prueba Funcional 13.....	176
Tabla 131: Prueba Funcional 14.....	177
Tabla 132: Prueba Funcional 15.....	178
Tabla 133: Prueba Funcional 16.....	179
Tabla 134: Prueba Funcional 17.....	180
Tabla 135: Prueba Funcional 18.....	181
Tabla 136: Matriz RSF – PF.....	183
Tabla 137: Prueba Estrés 1.....	184



Tabla 138: Prueba Estrés 2. ....	184
Tabla 139: Prueba Estrés 3. ....	185
Tabla 140: Prueba Estrés 4. ....	185
Tabla 141: Fases del proyecto y duración. ....	190
Tabla 142: Horas trabajadas por fase de proyecto. ....	193
Tabla 143: Coste personal. ....	194
Tabla 144: Coste en Hardware. ....	195
Tabla 145: Coste total. ....	197



## **CAPÍTULO 1: INTRODUCCIÓN Y OBJETIVOS.**

A continuación, analizaremos las motivaciones y objetivos que nos han empujado a realizar el proyecto Campus Life, analizando paso a paso su desarrollo desde el análisis hasta las pruebas, pasando por el diseño y la implementación. Para que sea más sencilla su comprensión se realizará un breve resumen al inicio de cada apartado que dará una visión general del tema a tratar.

## 1.1 VISIÓN GENERAL.

La principal necesidad que debe de cubrir la aplicación Campus Life es la gestión de gran parte de los trámites que se pueden realizar en la web del departamento de ESES (Espacio Estudiantes) trasladados al entorno de dispositivos móviles iOS. Se pueden realizar consultas e inscripciones sobre actividades deportivas, culturales y de orientación.

La principal tarea que realizará la aplicación es permitir al personal universitario inscribirse en las actividades que oferta ESES y poder visualizar dicha oferta por campus, también se podrán comprar de manera rápida y sencilla las diferentes tarjetas y abonos que se ofertan, además se incluirán diferentes servicios para la gestión y consulta de promociones, contacto y actividades favoritas.

## 1.2 MOTIVACIÓN.

En el mundo actual, se está viviendo una etapa de desarrollo abrumadora en el ámbito de las nuevas tecnologías y las comunicaciones. Este cambio tan significativo hace necesario actualizar todos los servicios que tenemos y los nuevos que se generan de una manera rápida y continua. No estar al día en los nuevos servicios, prestaciones o tendencias, puede desencadenar un abandono masivo por parte de los clientes que utilizan nuestros productos y como consecuencia un fracaso para nuestros proyectos.

La actual crecida en las ventas de smartphones o dispositivos móviles inteligentes, denota que hoy en día el uso del teléfono móvil es mucho más dinámico y ofrece una gran cantidad de posibilidades acercándose muy de cerca a los servicios que puede ofrecer un ordenador portátil o de sobremesa y en algunos casos superando en potencia y prestaciones a estos últimos.

Por todo ello el desarrollo de aplicaciones móviles esta siendo la nueva tendencia en el ámbito del desarrollo. Migrar todos esos servicios que anteriormente se ofrecían a los usuarios vía navegador o cliente de escritorio y ahora deben de ser optimizados a las necesidades que ofrecen los nuevos teléfonos inteligentes.

Así que la principal motivación para realizar este proyecto es la de actualizar los servicios que ofrece ESES vía web a un entorno en auge como el de las aplicaciones móviles, facilitando al usuario la realización de gestiones desde cualquier plataforma, esto además nos permite llegar a un mayor número de usuarios, actualizarnos y estar al día en las últimas tendencias tecnológicas.

### 1.3 OBJETIVOS.

El principal objetivo de la aplicación Campus Life es intentar trasladar todas las posibilidades que ofrece el portal web de Espacio Estudiantes intentando mejorar la usabilidad de los usuarios añadiendo funcionalidades nativas de nuestros dispositivos móviles inteligentes, además se pretende facilitar la realización de las tareas pertinentes desde cualquier lugar y en cualquier momento gracias a la conectividad que ofrecen estos dispositivos.

Como objetivos inmediatos se pretende:

- Crear una aplicación nativa para iOS eliminando la dependencia del navegador y aprovechando funcionalidades específicas de los smartphones.
- Diseñar una aplicación rápida y sencilla de utilizar para todos los usuarios.
- Crear una aplicación que sea lo más ligera posible y que no consuma demasiados recursos, para que cualquier tipo de dispositivo móvil por escasas que sean sus características pueda usarla.
- Ha de permitir realizar compras, de manera sencilla y segura.
- Visualizar toda la oferta de actividades de todos los campus.
- Debe de permitir realizar inscripciones en actividades.
- Ofertar promociones y descuentos.

- Controlar los errores de usuario para minimizar los fallos producidos por usos inadecuados y evitar así frustraciones en el usuario.

#### 1.4 FASES DEL DESARROLLO.

El proyecto se dividirá en seis fases de desarrollo.

La primera fase trata de estudiar cuales serán las tecnologías, productos similares y plataformas de desarrollo para comprobar la viabilidad del proyecto. Tras un análisis meticuloso se decidirá la plataforma a utilizar, su arquitectura y sus características.

Una vez escogida la plataforma entramos en la fase dos, con el análisis de los requisitos que debe cumplir el sistema, para ello se realizarán varias reuniones con el departamento de Espacio Estudiantes y se analizarán las necesidades, la viabilidad de los requisitos y la prioridad de los mismos.

Cuando finalicemos la lista de requisitos, entraremos en la fase tres, dedicada al diseño de la arquitectura, indicando que componentes cubren cada necesidad de los requisitos.

La cuarta fase del desarrollo es la implementación del código de la aplicación, es fundamental que las fases anteriores estén cerradas y bien diseñadas pues un cambio en cualquiera de ellas arrastraría un error que resultaría muy complicado de corregir, retrasando la entrega del proyecto y disparando sus costes.

La quinta fase es la fase de pruebas y puede solaparse con la fase de implementación, es muy aconsejable realizar pruebas según se realizan implementaciones de cada funcionalidad puesto que dejar las pruebas para el final de la implementación puede conllevar correcciones muy complejas de código, aun así existen pruebas que necesitan de la implementación completa para poder realizarse.

La última fase consiste en la elaboración del presente documento que explica de manera detallada todo el proceso de creación de la aplicación y que es

necesario para el lanzamiento de nuevas versiones y el estudio por parte de nuevos componentes en el equipo de creación.

## **1.5 ESTRUCTURA DEL DOCUMENTO.**

A continuación se detallará brevemente la finalidad de cada uno de los capítulos en los que se divide el documento.

### **1.5.1 CAPÍTULO 1, Introducción y objetivos.**

Su finalidad es dar una visión general del proyecto, cuáles han sido los objetivos y las motivaciones necesarias para emprender el proyecto, de esta manera comprenderemos mejor cada uno de los siguientes capítulos.

### **1.5.2 CAPÍTULO 2, Estado de la cuestión.**

Su objetivo es realizar un análisis de las diferentes tecnologías y plataformas existentes consensuando cual es la opción que mejor se adapta a las necesidades de nuestro proyecto y eligiendo sobre cual se desarrollará el mismo.

### **1.5.3 CAPÍTULO 3, Análisis.**

Su intención es realizar un proceso profundo en el análisis de requisitos, requisitos que nuestra aplicación debe de cumplir para satisfacer todas las necesidades acordadas con Espacio Estudiantes en la elaboración de los mismos, así como los casos de uso que se pueden producir en la aplicación.

### **1.5.4 CAPÍTULO 4, Diseño.**

Su propósito es definir el diseño de la aplicación adaptada a los requisitos acordados, su arquitectura y componentes utilizados, la base de datos y sus estructuras de almacenamiento y su interfaz de cliente que será lo que en definitiva el usuario final pueda ver.

#### **1.5.5 CAPÍTULO 5, Implementación y Pruebas.**

Su finalidad es realizar un análisis de la implementación, no se pretende mostrar el código de la aplicación sino más bien el tipo de diseño que se ha utilizado al programar y la estructura lógica, así como las utilidades que nos han ayudado en el proceso. También se realizarán pruebas de testeo para comprobar el correcto funcionamiento de la aplicación.

#### **1.5.6 CAPÍTULO 6, Planificación y presupuesto.**

Su objetivo es realizar un estudio del proceso y planificación del proyecto calculando cuales han sido los gastos ocasionados por la compra de material, licencias y sueldos de trabajadores, así como sobre costes o gastos extra, se debe de calcular las horas dedicadas a cada fase del proyecto y realizar un diagrama que muestre la línea de producción del proyecto.

#### **1.5.7 CAPÍTULO 7, Conclusiones y próximas novedades.**

Su propósito es llegar a una conclusión subjetiva y objetiva del trabajo realizado durante las diferentes fases del proyecto así como de las futuras funcionalidades que podrían desarrollarse en las nuevas versiones y que completarían o mejorarían los servicios prestados.





## CAPÍTULO 2. ESTADO DE LA CUESTIÓN.

El capítulo dos trata de analizar la actualidad tecnológica, así como las nuevas tendencias para determinar un enfoque adecuado del proyecto. Según las tendencias actuales podemos crear una aplicación preparada para el mercado contemporáneo y sus necesidades pero hay que profundizar en las tecnologías con alta probabilidad de prosperar en un futuro y crear una aplicación modulable y escalable que pueda evolucionar satisfactoriamente.

## 2.1 EVOLUCIÓN DE LA TECNOLOGÍA.

Las nuevas tecnologías aplicadas al ámbito del deporte, las actividades culturales y los viajes son numerosas, han evolucionado al ritmo que surgían los nuevos dispositivos inalámbricos y con conectividad, el internet de las cosas a proporcionado conexión en situaciones hasta antes impensables.

Este nuevo abanico de posibilidades, ha abierto el sector al desarrollo de aplicaciones especializadas en servicios puntuales. Donde antes una persona tenía una ventana abierta a la red y los datos almacenados únicamente en un dispositivo fijo, ahora los dispositivos se mueven con nosotros y nos monitorizan obteniendo datos y mostrándonos información allá donde vayamos en tiempo real y todo gracias a la conectividad y el almacenamiento en la nube.

Este potencial es aprovechado por todos los sectores para ganar cuota de mercado desarrollando aplicaciones para todo tipo de dispositivos, perder la posibilidad de desarrollar una aplicación para algún tipo de dispositivo móvil puede suponer una pérdida de mercado considerable. Por ello Campus Life intenta adaptarse a los tiempos y no perder a esos usuarios que conectan con sus dispositivos en cualquier momento y en cualquier lugar cumpliendo con sus necesidades en el menor tiempo posible.

### 2.1.1 ORIGEN.

Analizando la tecnología desde el origen, básicamente cualquier usuario que quisiera apuntarse a una actividad tenía que ir físicamente a los puntos de información y oficinas de Espacio Estudiantes, estudiar las diferentes ofertas y elegir una actividad que se adaptase a sus necesidades, en caso de no decidirse, tenía que volver en otro momento, es decir que en la mayoría de casos el usuario se desplazaba 2 veces como mínimo, una para informarse y otra para inscribirse.

Es en este momento cuando el usuario perdía más tiempo para hacer un trámite, las aplicaciones únicamente se desarrollaban para los empleados del sector, es decir solo unos pocos podían acceder al servicio y los usuarios tenían que acudir a ellos para poder tramitar sus inscripciones.

### 2.1.2 INTERNET.

Con la revolución de internet y las redes, algunos usuarios podían acceder a páginas estáticas donde visualizar toda la oferta sin tener que desplazarse de casa, en este momento hay una ventana que acerca el servicio al usuario al menos en lo que a la parte informativa se refiere, aún así el usuario tenía que desplazarse a los centros para tramitar su inscripción puesto que las aplicaciones seguían siendo accesibles únicamente para los empleados y los usuarios necesitaban acudir a ellos para poder tramitar sus inscripciones. La mejora afecta mayoritariamente al usuario y crea un boom de desarrollo web donde las empresas ofertaban sus servicios, aún así la cantidad de usuarios que tenían acceso a la red era mínimo.

### 2.1.3 WEB 2.0.

Con la entrada de la web 2.0 surgieron nuevas aplicaciones web que automatizaban las tareas permitiendo al usuario interactuar con el servicio eliminando así mucho trabajo a los empleados y sobre todo permitiendo al usuario realizar gestiones desde su domicilio, eligiendo con más calma la oferta adecuada a sus necesidades. Comienzan a diseñarse aplicaciones web que realizan funcionalidades del cliente que antes únicamente podían realizar los empleados.

### 2.1.4 MÓVILES Y SMARTPHONES.

Con la llegada de la nueva generación de dispositivos móviles con conexión a internet y el desarrollo de los nuevos sistemas operativos, el potencial de los smartphones prácticamente iguala a los dispositivos fijos del hogar y supera con creces a los antiguos ordenadores, este estallido de potencia junto con la calidad de conexión que ofrecen los ISP dan opción a crear aplicaciones con las mismas funcionalidades que podemos encontrar en los dispositivos fijos conectados a una red de cable.

En esta nueva generación nos encontramos nosotros ahora mismo y Campus Life aprovecha todo el potencial de los smartphones para cargar gran cantidad de imágenes y videos, realizar inscripciones en el momento, geo-localizarnos en un mapa y mostrarnos las oficinas de cada campus o realizar llamadas desde la aplicación, el usuario puede interactuar y disfrutar de los servicios

aprovechando los beneficios y particularidades que ofrecen los nuevos dispositivos inteligentes.

Con respecto a las tecnología deportivas y culturales en el ámbito de la educación se está trabajando actualmente en ello, hay pocas universidades que puedan desarrollar aplicaciones para todos los servicios que tienen implementados en la web y básicamente se dedican a desarrollar aquellas con más potencial y más funcionales que arrastran a un número mayor de usuarios.

Sin embargo todas ellas tienen sus servicios disponibles en la web lo cual nos sitúa en un tránsito entre las tecnologías web y las tecnologías móviles, pero aunque el progreso sea impresionante, el desarrollo es costoso y largo, y seguro que dentro de unos años estaremos disfrutando de todos los servicios que podamos imaginar en nuestros smartphones.

## 2.2 ANÁLISIS DE LA PLATAFORMA MÓVIL.

A día de hoy la oferta que puede elegir un usuario de smartphones es muy amplia, cada uno de estos dispositivos viene por defecto con un sistema operativo que tiene ligado al hardware, así que cuando un consumidor se decide por realizar la compra de uno de ellos no solo debe mirar las especificaciones técnicas referentes al hardware, sino que también debe de percatarse de cuál es el sistema operativo instalado puesto que elegir entre uno u otro puede ser decisivo a la hora de analizar el rendimiento y las posibilidades que nos ofrece ambos. En la actualidad las plataformas móviles con más expansión de mercado son, los sistema Android e iOS seguidos más de lejos por Windows Phone, BlackBerry o Firefox OS.

Para saber un poco más acerca de cada una de ellas, pasaremos a analizarlas brevemente.

### 2.2.1 ANDROID.

Android es un sistema operativo basado en Linux, originalmente diseñado para cámaras fotográficas, muchos lo conocen como el sistema operativo de Google, pero esto ocurrió después de que la empresa Android.Inc, fundada en 2003 y comprada por Google en el 2005, anunciase en 2007 el sistema operativo y lo lanzase en 2008 al mercado. Google decidió modificarlo para utilizarlo en smartphones y tablets, y su funcionalidad no termina aquí, actualmente se encuentra en proceso de desarrollo para poder utilizarse en Netbooks y PCs. Es sin lugar a duda el SO para dispositivos móviles líder del mercado.

Android fue desarrollado en Java y es de código abierto lo cual implica que su código se debe poner al alcance de todos y que todos podremos hacer con este código lo que nos parezca oportuno, modificarlo, ampliarlo, recortarlo, pero

siempre estaremos en la obligación de volver a distribuirlo con la misma licencia. Además posee una amplia comunidad de desarrolladores que aporta una gran solidez al sistema y a sus aplicaciones, así como a documentación y ejemplos.



Como se ha comentado anteriormente Android fue liberado en 2008 y en tan solo 2 años, a finales de 2010 ya era el líder de mercado de SSOO para dispositivos móviles desbancando a Symbian de la Empresa Nokia que había liderado el sector durante años.

En la actualidad hay en el mundo más de 1400 millones de dispositivos que utilizan Android ,casi un 70% de mercado y las previsiones son buenas para este SO que se utiliza la mayoría de empresas del sector, como Samsung, HTC, Sony, Huawei, LG o Motorola.

### 2.2.2 iOS.

iOS, originalmente denominado iPhone OS fue el sistema operativo desarrollado por la empresa Apple para sus dispositivos móviles iPhone como su nombre indica. Además se dice que estos fueron los primeros teléfonos inteligentes que gobernaron el mercado de los móviles dando luz a una nueva generación de dispositivos. iOS fue lanzado por primera vez en 2007, un año antes que Android, para los iPhone aunque poco después se utilizaría para más productos de la empresa como los iPad, iPod Touch o el Apple TV. Se dice que es el SO mejor optimizado en cuanto a recursos del dispositivo y rendimiento se refiere.

Su principal ventaja como se ha dicho antes es su optimización y su simplicidad, haciéndolo sencillo y cómodo para el usuario, pero a todo ello hay que añadir la estética y el diseño utilizado siempre por Apple para sus dispositivos, que es sin duda alguna uno de los mayores atractivos, sin embargo es un sistema cerrado que solo podemos utilizar en dispositivos Apple lo cual cierra el mercado de dispositivos que podemos obtener con este sistema.

Aun así Apple tiene una cuota de mercado muy amplia y es el segundo líder por detrás de Android llegando casi al 18% de cuota de mercado. En el segundo trimestre de 2016 se vendieron más de 51 millones de dispositivos iPhone y aunque parezcan muchos, es la primera vez que bajan las ventas desde que se comercializan los teléfonos de Apple.



Ilustración 1: iOS 9

### 2.2.3 WINDOWS PHONE.

Windows Phone, como su nombre indica, es el sistema operativo desarrollado por la empresa Microsoft para sus dispositivos móviles. Windows Phone está basado en el sistema operativo Windows CE y fue originalmente creado para un mercado profesional bajo el nombre de Windows Mobile. En su origen no tuvo mucha aceptación y era bastante diferente a sus principales competidores, está diseñado estéticamente de forma similar a las versiones de escritorio de Windows, pero Microsoft ha sabido adaptarse y obtener puntos fuertes como un gran mercado de aplicaciones, cerca de 120.000 disponibles en la Windows Phone Store, además de su compatibilidad con muchas aplicaciones desarrolladas para Android.

Actualmente, ocupa la tercera posición en cuanto a cuota de mercado con un 11% aproximadamente, pero muy lejos de los dos gigantes Android e iOS.

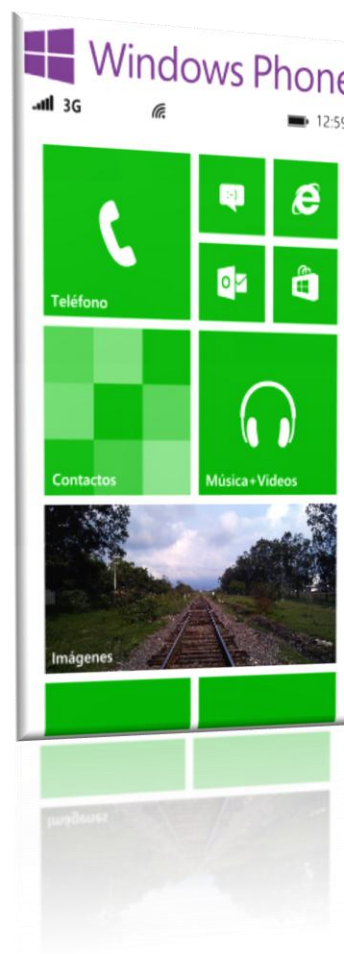


Ilustración 2: Windows Phone.



#### 2.2.4 BLACKBERRY.

BlackBerry es un sistema operativo para móviles desarrollado por Research In Motion y presentado en el WES 2010. Este sistema operativo tiene los mismos defectos que iOS y es que solo lo encontramos en dispositivos de la marca BlackBerry pero a diferencia de Apple, BlackBerry tuvo un declive con sus dispositivos, apostó por dispositivos enfocados al ámbito empresarial y con un teclado físico, lo cual hacía más pequeña la dimensión de sus pantallas y reducía el abanico de posibilidades.

Actualmente BlackBerry está intentando remontar con teléfonos táctiles, pero el mercado empresarial es más reducido y los dos gigantes se han comido su parte del mercado en este sector, lo cual reduce el número de dispositivos vendidos y por consecuencia el uso de su sistema operativo.



**Ilustración 3: BlackBerry OS.**

### 2.2.5 SYMBIAN.

Symbian es sin duda el gran derrotado, al igual que BlackBerry, no supo ver las necesidades del mercado y se estancó en su proyecto, pero a diferencia de BlackBerry, la caída fue mucho mayor puesto que Symbian era líder del mercado en dispositivos móviles. Fue producto de la alianza de varias empresas , y entre las más destacadas encontramos a Nokia como la más importante, Samsung, Lenovo, Sony Ericsson, BenQ, Fujitsu, Siemens, LG, Motorola. Se podría decir que estaba en la cima copando casi todos los dispositivos que se fabricaban.

Symbian se encuentra en declive y su cuota de mercado es despreciable y pertenece a dispositivos antiguos, que podríamos denominar obsoletos, pero que mucha gente sigue utilizando y en países subdesarrollados son todavía de los más usados.

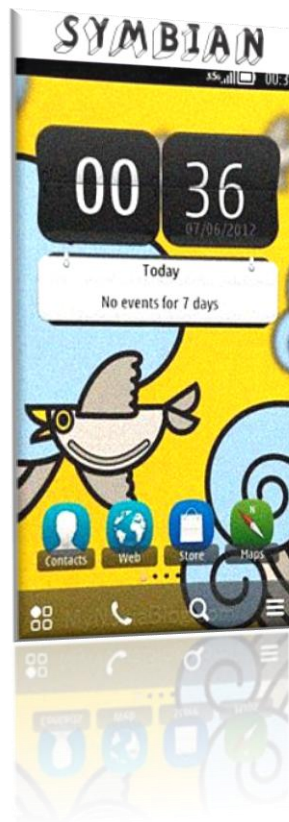


Ilustración 4: Symbian OS.

### 2.2.6 FIREFOX OS.

Firefox OS es un sistema operativo móvil desarrollado por Mozilla Corporation para teléfonos inteligentes y tabletas, está basado en HTML5 con un núcleo Linux. El sistema se enfoca especialmente en los dispositivos móviles de gama baja, trata de abarcar el mercado de los dispositivos con poco rendimiento permitiendo a las aplicaciones HTML5 comunicarse directamente con el hardware del dispositivo usando JavaScript y Open Web APIs.

Mozilla intenta ofrecer algo nuevo e intentar conquistar el mercado de los dispositivos low cost, pero Android tiene la totalidad de este mercado y las mejoras que proporciona Firefox OS no son suficientes como para dar un vuelco. Además Google cuenta con un gran presupuesto y el desgaste sufrido por Mozilla es muy grande para el poco beneficio obtenido.



Ilustración 5: FireFox OS.

### 2.2.7 UBUNTU TOUCH.

Ubuntu Touch es un sistema operativo móvil basado en Linux. Es desarrollado por Canonical Ltd.. Básicamente es un reflejo de lo ocurrido con Firefox OS, al igual que Android son de código abierto y tienen todas las ventajas del campo de los desarrolladores, pero Android es un gigante y tanto Firefox OS como Ubuntu Touch no ofrecen nada novedoso que pueda dar un mordisco al mercado de Android, por lo que el gigante sigue aumentando sus ventas con un bajo esfuerzo en dispositivos de gama baja, mientras que sus competidores tienen que realizar un gran esfuerzo para conseguir unos resultados casi nulos.

Aun con estos desgarradores resultados, Firefox OS y Ubuntu Touch están en auge en el mundo de los desarrolladores y son proyectos jóvenes con aires frescos e ideas nuevas que podrían dar la sorpresa en cualquier momento, mientras que BlackBerry y Symbian están en declive más cerca de su desaparición con proyectos anticuados y sobre una base obsoleta.

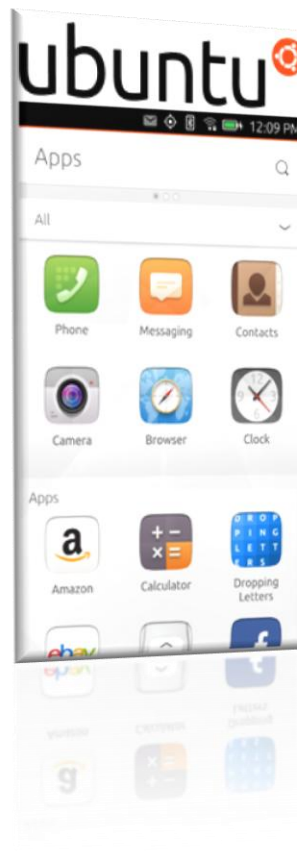
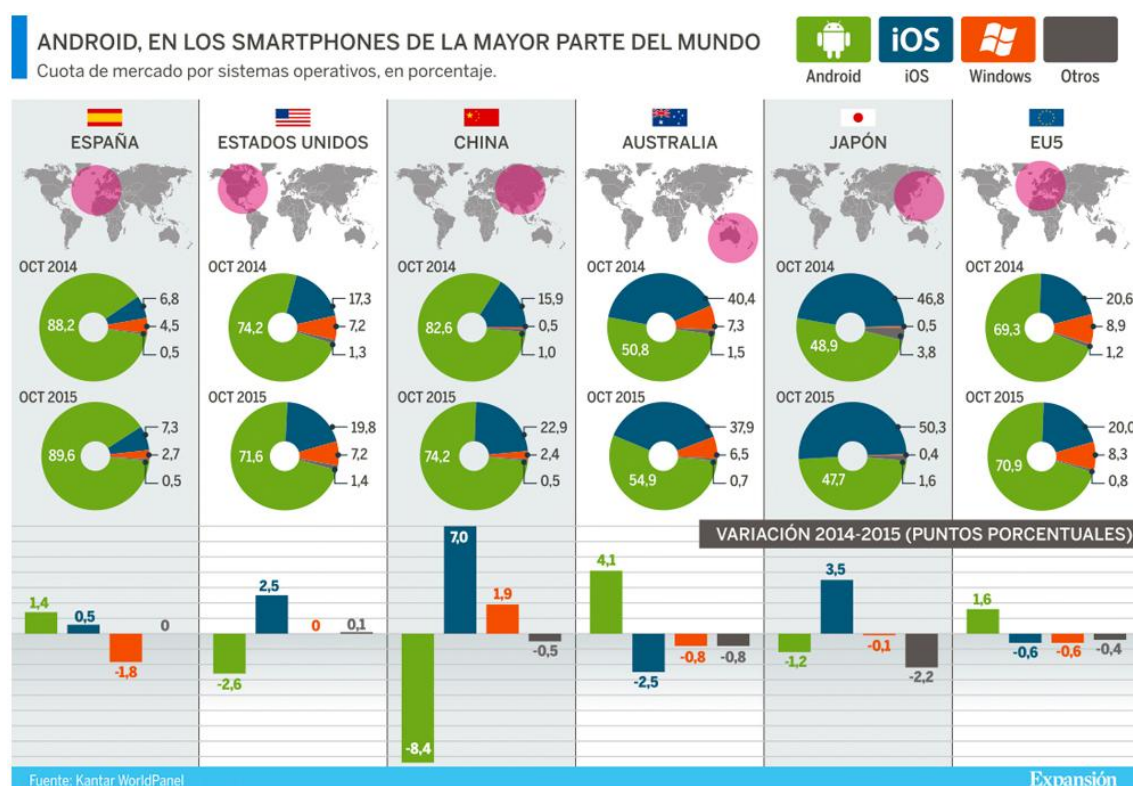


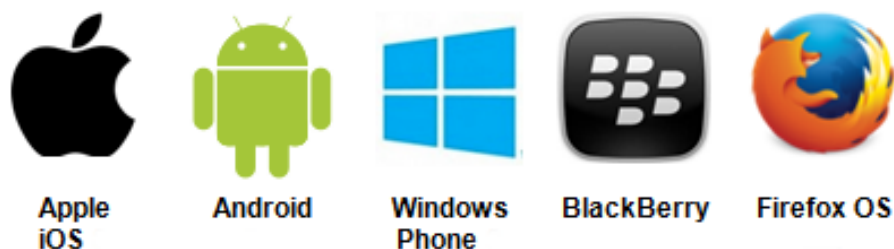
Ilustración 6: Ubuntu Touch.

En las siguientes imágenes podemos observar las cuotas de mercado de los diferentes operadores tanto a nivel mundial como a nivel nacional así como por países más desarrollados del mundo en cuando a dispositivos inteligentes se refiere.



**Ilustración 7: Cuota de mercado de los smarthones en diferentes regiones.**

En cuanto a la segunda imagen se refiere, se muestra una tabla que nos indica cuales son las características de cada fabricante y algunas de sus propiedades como pueden ser su licencia, CPU, arquitectura, fecha de lanzamiento, núcleo del sistema operativo, etc.



Compañía	Apple	Open Handset Alliance	Microsoft	BlackBerry	Mozilla Foundation
Núcleo del SO	Mac OS X	Linux	Windows NT	QNX	Linux
Licencia de software	Propietaria	Libre y abierto	Propietaria	Propietaria	Libre y abierto
Año de lanzamiento	2007	2008	2010	1999	2013
Fabricante único	Sí	No	No	Sí	No
Variedad de dispositivos	Modelo único	Muy alta	Media	Baja	Muy baja
Soporte memoria externa	No	Sí	Sí	Sí	Sí
Motor del navegador web	WebKit	WebKit/Chromium (Blink)	Trident	WebKit	WebKit
Tienda de aplicaciones	App Store	Google Play	Windows Marketplace	BlackBerry World	Firefox Marketplace
Número de aplicaciones	800.000 (marzo 2013)	800.000 (marzo 2013)	130.000 (enero 2013)	100.000 (enero 2013)	¿?
Coste publicar	\$99 / año	\$25 una vez	\$99 / año	Sin coste	Sin coste
Otras tiendas sin supervisión	No	Si	No	Si	Si
Familia CPU soportada	ARM	ARM, MIPS, x86	ARM	ARM	ARM, x86
Soporte 64 bits	Si	Si	No	No	No
Máquina virtual	No	Dalvik / ART	.net	No	Navegador Web
Lenguaje de programación	Objective-C, C++	Java, C++	C#, Visual Basic, C++	C, C++, Java	HTML5, CSS, JavaScript
Plataforma de desarrollo	Mac	Windows, Mac, Linux	Windows	Windows, Mac	Windows, Mac, Linux
Multiusuario	No	Si	No	No	No
Modo invitado	Si	Si	No	No	No

Ilustración 8: Características de los principales Sistemas Operativos del mercado.



## **2.3 CRITERIOS DE SELECCIÓN DEL SISTEMA OPERATIVO.**

Hemos analizado anteriormente los 7 principales sistemas operativos que ocupan el mercado, y debemos tomar una decisión crucial en el desarrollo de la aplicación Campus Life, ¿Para qué sistema operativo debemos desarrollar nuestra aplicación?

Pero antes de tomar una decisión debemos de analizar las necesidades de nuestra aplicación y el rendimiento que nos debe de prestar el dispositivo.

### **2.3.1 PRIMERA NECESIDAD.**

Se trata de una aplicación de ámbito cerrado, es decir, todo el mundo puede descargarla pero solo pueden acceder a ella la comunidad de usuarios de la Universidad Carlos III de Madrid, con esto sabemos que el número de usuarios no será muy elevado pero la población que descargue la aplicación será una muestra real del sistema en que vivimos, alumnos con acceso a móviles de gama media-alta, profesores con posibilidades de tener dispositivos populares y potentes.

En general se presupone que la gran mayoría de los usuarios que utilicen Campus Life poseerán un dispositivo lo suficientemente potente como para mover la aplicación sin problemas.

### **2.3.2 SEGUNDA NECESIDAD.**

La aplicación tiene múltiples funciones que necesitan de una amplia pantalla para ver con claridad todos los detalles y poder acceder a ellos, como por ejemplo los mapas de localización o los videos de las actividades.

### **2.3.3 TERCERA NECESIDAD.**

La aplicación sufrirá a lo largo de su existencia varias actualizaciones y mejoras, aumentando el procesamiento y adaptándose a las nuevas tecnologías emergentes, cada cambio de versión podría hacer que dejaran de funcionar algunos servicios dependiendo del dispositivo y de la versión del mismo.

Por ellos el sistema operativo elegido debe de estar en constante evolución y mejorando para poder adaptarse a los cambios y mejoras realizados.

## 2.4 CONCLUSIÓN SOBRE LA ELECCIÓN DEL SISTEMA OPERATIVO.

Estas tres necesidades son fundamentales para elegir el sistema operativo en el que se podrá ejecutar nuestra aplicación. Y a raíz de ellos se tomaran unas conclusiones.

### 2.4.1 PRIMERA CONCLUSIÓN.

Está claro que necesitamos un sistema operativo en el que ejecutar nuestra aplicación, pero casi siempre las aplicaciones se desarrollan para varios sistemas operativos con la idea de cubrir el máximo número de dispositivos del mercado.

Podríamos analizar varios aspectos de los dispositivos como por ejemplo el rendimiento, el diseño o el tamaño de pantalla, pero observando claramente las cuotas de mercado, está claro que entre Android e iOS copan más del 90% del mercado, además analizando más específicamente en la zona de Madrid (España), que es donde se presupone se descargara y utilizara la mayoría de veces.

Analizando los resultados estadísticos, se deduce que las cuotas son todavía mayores que la media mundial llegando casi a un 95% de mercado, además mirando las diferencias entre los dos últimos años, tanto los sistemas Android como los sistemas iOS han aumentado su cuota ligeramente ganándole terreno a los sistemas menos populares.

Tras esta interesante conclusión decimos sin lugar a duda que el desarrollo de la aplicación debe realizarse tanto para Android como para iOS, podríamos pensar que realizando un único desarrollo para Android tendríamos unos grandes resultados ya que llegaríamos a casi un 90% de usuarios, pero sería un error puesto que los usuarios de iOS no son muy populares en España pero en países como Australia y Japón tienen una cuota de mercado de casi el 50% y como la Universidad Carlos III tiene un amplio número de usuarios internacionales, no podemos pasar por alto este detalle.

Para terminar, hemos decidido que el desarrollo para que la aplicación pueda ser ejecutada en sistemas Android e iOS es fundamental para el éxito de nuestra aplicación. Cabe destacar que únicamente hemos analizado las cuotas de mercado para tomar esta decisión.



#### 2.4.2 SEGUNDA CONCLUSIÓN.

Nos quedan por analizar 5 de los sistemas vistos anteriormente. Sabemos que la aplicación será desarrollada para ejecutarse en dispositivos inteligentes, podrá correr en tablets pero su diseño no estará optimizado al 100% para estos dispositivos.

Uno de los aspectos más importantes es los precedentes de los sistemas que aún nos quedan por analizar, como bien dijimos en el apartado anterior hay una clara diferencia entre los sistemas BlackBerry y Symbian con respecto a Firefox OS y Ubuntu Touch y es que mientras que los primeros se encuentran claramente en declive los segundos se encuentran en fases tempranas de desarrollo y son relativamente nuevos, además los primeros únicamente son capaces de ejecutarse en sus propios dispositivos lo cual hace más inviable su desarrollo, y para terminar, las expectativas de futuro no son muy alentadoras.

Queda claro que tanto BlackBerry como Symbian no cumplen ningún requisito para poder emprender un desarrollo en estos sistemas, los dispositivos en los que se ejecutan son pobres en potencia, las pantallas de estos dispositivos suelen ser las más pequeñas del mercado, el software no tiene un futuro prometedor y las cuotas de mercado son casi nulas en el ámbito donde se pretende utilizar Campus Life.

Por todo ello BlackBerry OS y Symbian quedan descartados para su posible desarrollo.

#### 2.4.3 TERCERA CONCLUSIÓN.

Firefox OS y Ubuntu Touch, como se dijo anteriormente son dos proyectos jóvenes e innovadores con grandes ideas, pero ¿Realmente merece la pena embarcarse en el desarrollo para estas plataformas?

Basándonos en cuotas de mercado y aunque parezca repetitivo el tema, es uno de los aspectos más importantes, no merece la pena. Sus porcentajes están a la par que la de BlackBerry e incluso son más bajos que los de Symbian, puede que en un futuro el crecimiento de estos dos sistemas se dispare y merezca la pena comenzar el desarrollo para ellos, pero a día de hoy los costes de producción superan claramente el beneficio obtenido.

Otro dato más, es que los dispositivos donde se ejecutan estos sistemas son terminales low-cost de gama baja, lo cual no concuerda con la muestra de usuarios que utilizarán la aplicación Campus Life.

Por todo ello Firefox OS y Ubuntu Touch quedan descartados para su posible desarrollo, pero no se descarta comenzar posteriormente un desarrollo para estos sistemas.

#### 2.4.4 CUARTA CONCLUSIÓN.

Quizás es la más complicada de analizar, puesto que el último sistema que nos queda por analizar (Windows Phone) cumple relativamente bien los requisitos para su desarrollo, pero tiene algunas carencias importantes.

En principio Windows Phone tiene una cuota de mercado en España similar a iOS, quizá un poco inferior pero no tan notable como para descartarlo, sin embargo analizando fuera de nuestro país las cuotas se mantienen, mientras que iOS crece en la gran mayoría. Además las estadísticas muestran un dato sorprendente y es que mientras los dos grandes sistemas crecen cada año arañando mercado a sus competidores, Windows Phone no termina de despegar, incluso notando descensos significativos.

Otro aspecto importante es la gran dependencia que tiene Windows Phone de los dispositivos Lumia y es que al igual que iOS depende de las ventas de iPhone, parece que Windows Phone depende demasiado de las ventas de los Lumia que además parecen no tener un futuro muy claro, toda esta incertidumbre hace tambalear el posible desarrollo de Campus Life.

Por todo ello Windows Phone queda descartado para su posible desarrollo, pero no se descarta comenzar posteriormente un desarrollo para este sistema.

#### 2.4.5 ELECCIÓN DEFINITIVA, iOS.

Campus Life está disponible también para Android y tiene gran cantidad de puntos en común con el desarrollo de Campus Life para iOS, pero en este documento nos centraremos en el desarrollo exclusivo de la aplicación para iOS.

La aplicación Campus Life está diseñada originalmente para ejecutarse en dispositivos iPhone, a pesar de que también puedan correr en otros dispositivos de la compañía como puedan ser los iPad, aun así no se asegura el correcto funcionamiento en estos dispositivos debido a que no fue implementada para ejecutarse en ellos y además tienen algunas diferencias técnicas que pueden entorpecer el correcto funcionamiento de la aplicación.

Podemos observar las especificaciones técnicas de los últimos modelos de dispositivos donde se puede ejecutar la aplicación Campus Life, cabe destacar que originalmente el proyecto debería de implementarse sobre iOS 8, pero con la llegada del nuevo sistema iOS 9 y la venta de los nuevos dispositivos se ha actualizado las especificaciones y se desarrollará para iOS 9, además se dará cobertura a aquellos dispositivos que tengan instalada hasta la versión 7.1.

					
	iPhone 6S Plus	iPhone 6S	iPhone 6 Plus	iPhone 6	iPhone SE
	<a href="#">Comprar</a>	<a href="#">Más Info</a>	<a href="#">Comprar</a>	<a href="#">Comprar</a>	<a href="#">Comprar</a>
<b>Capacidad<sup>1</sup></b>	16 GB 64 GB 128 GB	16 GB 64 GB 128 GB	16 GB 64 GB	16 GB 64 GB	16 GB 64 GB
<b>Pantalla</b>	Retina HD con 3D Touch 5,5 pulgadas (en diagonal) Resolución de 1.920 por 1.080 a 401 p/p	Retina HD con 3D Touch 4,7 pulgadas (en diagonal) Resolución de 1.334 por 750 a 326 p/p	Retina HD 5,5 pulgadas (en diagonal) Resolución de 1.920 por 1.080 a 401 p/p	Retina HD 4,7 pulgadas (en diagonal) Resolución de 1.334 por 750 a 326 p/p	Pantalla Retina 4 pulgadas (en diagonal) Resolución de 1.136 por 640 a 326 p/p
<b>Dimensiones y peso<sup>2</sup></b>	Alto: 15,82 cm Ancho: 7,79 cm Grosor: 0,73 cm Peso: 192 g	Alto: 13,83 cm Ancho: 6,71 cm Grosor: 0,71 cm Peso: 143 g	Alto: 15,81 cm Ancho: 7,78 cm Grosor: 0,71 cm Peso: 172 g	Alto: 13,81 cm Ancho: 6,7 cm Grosor: 0,69 cm Peso: 129 g	Alto: 12,38 cm Ancho: 5,86 cm Grosor: 0,76 cm Peso: 113 g
<b>Chip</b>	Chip A9 con arquitectura de 64 bits Coprocesador de movimiento M9 integrado	Chip A9 con arquitectura de 64 bits Coprocesador de movimiento M9 integrado	Chip A8 con arquitectura de 64 bits Coprocesador de movimiento M8	Chip A8 con arquitectura de 64 bits Coprocesador de movimiento M8	Chip A9 con arquitectura de 64 bits Coprocesador de movimiento M9 integrado
<b>Redes móviles e inalámbricas</b>	GSM/EDGE UMTS/HSPA+ DC-HSDPA CDMA EV-DO Rev. A  4G LTE Advanced <sup>3</sup> Wi-Fi 802.11a/b/g/n/ac con MIMO Bluetooth 4.2 GPS y GLONASS VoLTE <sup>3</sup> NFC	GSM/EDGE UMTS/HSPA+ DC-HSDPA CDMA EV-DO Rev. A  4G LTE Advanced <sup>3</sup> Wi-Fi 802.11a/b/g/n/ac con MIMO Bluetooth 4.2 GPS y GLONASS VoLTE <sup>3</sup> NFC	GSM/EDGE UMTS/HSPA+ DC-HSDPA CDMA EV-DO Rev. A y Rev. B (solo modelos CDMA)  4G LTE <sup>3</sup> Wi-Fi 802.11a/b/g/n/ac Bluetooth 4.2 GPS y GLONASS VoLTE <sup>3</sup> NFC	GSM/EDGE UMTS/HSPA+ DC-HSDPA CDMA EV-DO Rev. A y Rev. B (solo modelos CDMA)  4G LTE <sup>3</sup> Wi-Fi 802.11a/b/g/n/ac Bluetooth 4.2 GPS y GLONASS VoLTE <sup>3</sup> NFC	GSM/EDGE UMTS/HSPA+ DC-HSDPA CDMA EV-DO Rev. A  4G LTE <sup>3</sup> Wi-Fi 802.11a/b/g/n/ac Bluetooth 4.2 GPS y GLONASS VoLTE <sup>3</sup> NFC
<b>Touch ID</b>	Sensor de identidad por huella de 2.ª generación integrado en el botón de inicio	Sensor de identidad por huella de 2.ª generación integrado en el botón de inicio	Sensor de identidad por huella integrado en el botón de inicio	Sensor de identidad por huella integrado en el botón de inicio	Sensor de identidad por huella integrado en el botón de inicio
<b>Cámara iSight</b>	Cámara iSight de 12 Mpx con píxeles de 1,22 µ Apertura de f/2,2 Live Photos Estabilización óptica de imagen Flash True Tone Enfoque automático con Focus Pixels Control de exposición HDR automático para fotos Fotos panorámicas (hasta 63 megapíxeles) Estabilización automática Modo ráfaga	Cámara iSight de 12 Mpx con píxeles de 1,22 µ Apertura de f/2,2 Live Photos - Flash True Tone Enfoque automático con Focus Pixels Control de exposición HDR automático para fotos Fotos panorámicas (hasta 63 megapíxeles) Estabilización automática Modo ráfaga	Cámara iSight de 8 Mpx con píxeles de 1,5 µ Apertura de f/2,2 - Estabilización óptica de imagen Flash True Tone Enfoque automático con Focus Pixels Control de exposición HDR automático para fotos Fotos panorámicas (hasta 43 megapíxeles) Estabilización automática Modo ráfaga	Cámara iSight de 8 Mpx con píxeles de 1,5 µ Apertura de f/2,2 - - Flash True Tone Enfoque automático con Focus Pixels Control de exposición HDR automático para fotos Fotos panorámicas (hasta 43 megapíxeles) Estabilización automática Modo ráfaga	Cámara iSight de 12 Mpx con píxeles de 1,22 µ Apertura de f/2,2 Live Photos - Flash True Tone Enfoque automático con Focus Pixels Control de exposición HDR automático para fotos Fotos panorámicas (hasta 63 megapíxeles) Estabilización automática Modo ráfaga

**Ilustración 9: Especificaciones de los dispositivos elegidos para el desarrollo.**

## 2.5 ANÁLISIS DE APLICACIONES.

A continuación se analizan varias universidades españolas para comprobar aplicaciones similares a Campus Life y poder tener referencia de nuestra principal competencia. Comenzaremos analizando la oferta de aplicaciones de las universidades más cercanas de Madrid.

La primera y nuestra principal competencia por cercanía en la zona sur de Madrid es la universidad Rey Juan Carlos de Madrid que únicamente consta de una aplicación móvil Campus Virtual URJC, no aparece como una aplicación desarrollada por la universidad como corporación y sería más parecida a la aplicación Aula Global de la UC3M, además no tiene ningún contenido que permita realizar operaciones similares a Campus Life.

La universidad Complutense de Madrid consta de varias aplicaciones móviles, todas figuran como aplicaciones desarrolladas por la universidad como corporación, UCMóvil es similar a la aplicación UC3M, con acceso a la biblioteca pero sin ningún contenido que permita realizar operaciones similares a Campus Life. También cuenta con UCMatricula y EventosUCM, esta última si tiene una cierta similitud a Campus Life y sería una mezcla entre las aplicaciones de UC3MAgenda y Campus Life.

### UCM eventos

EventosUCM, permite ver los próximos eventos que se realizarán en la UCM, visualizar la localización del evento, marcarlo como evento favorito y recibir avisos sobre eventos. No tiene nada parecido a Ofertas, no se puede contactar con las oficinas del servicio y no se pueden realizar compras, en este sentido es mucho más una aplicación informativa que interactiva donde el usuario no tiene tanto control.

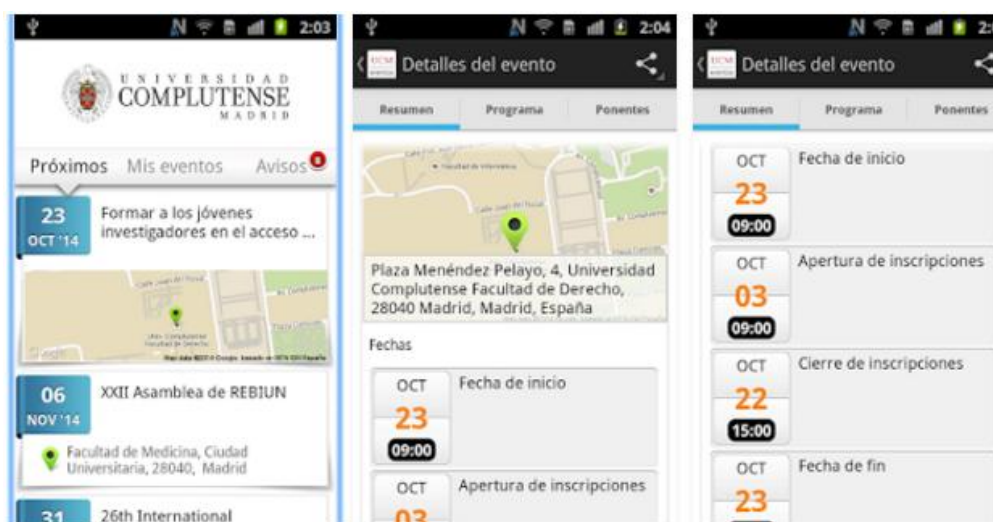


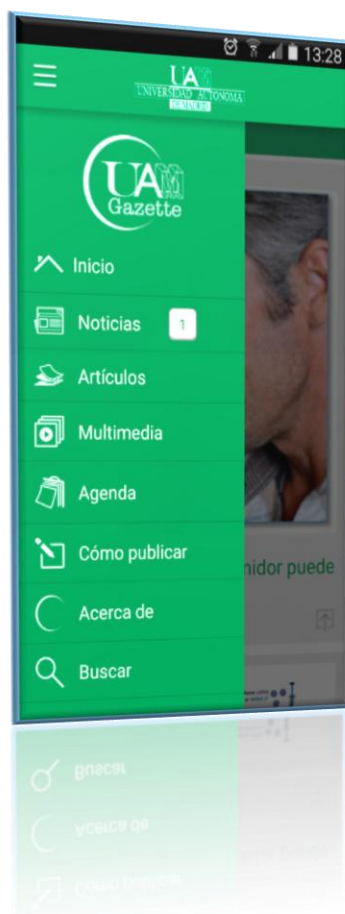
Ilustración 10: Universidad Complutense de Madrid.

Nuestra siguiente competidora es la Universidad Autónoma de Madrid y consta de varias aplicaciones para gestión de notas, PAU y encuestas, pero la aplicación que nos interesa a nosotros es UAM Gazzete.



UAM Gazzete es una aplicación enfocada a la divulgación científica y eventos científicos, no está enfocada al mismo ámbito que Campus Life pero su diseño tiene aspectos muy similares, tiene un menú lateral muy similar al de Campus Life, tiene elementos multimedia, pestañas y calendario aunque no posee compras ni localización. En general es una aplicación cuidada y con buenas referencias.

La oferta de aplicaciones por parte de la UAM es mucho menor a la UC3M y engloba en una sola aplicación muchas más características, es mucho más completa debido a la necesidad de suplir esa falta de aplicaciones, en general se trata de centralizar todo en una aplicación, aun así tienen falta de aplicaciones con más calado.



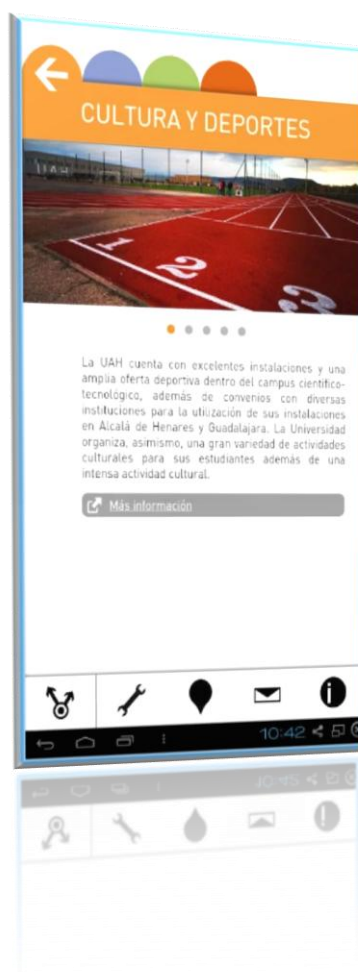
**Ilustración 11: Universidad Autónoma de Madrid.**



Continuamos con la Universidad Politécnica de Madrid, esta universidad tiene un número mucho menor de aplicaciones y se asemeja más a la URJC, únicamente posee 2 aplicaciones Estudiantes UPM y UPM Titulaciones, ninguna de ellas permite realizar operaciones similares a Campus Life, y se centran más en el ámbito educativo que en el cultural y deportivo además son prácticamente informativas y no poseen elementos de interacción como compras o localización.



Para terminar con la oferta de universidades públicas de Madrid analizaremos la universidad de Alcalá, una universidad que pese a poseer una única aplicación Universidad de Alcalá tiene bastante contenido y entre ellos una sección de deportes y cultura que podría encajar con la funcionalidad de Campus Life, aunque en principio la aplicación es meramente informativa cumple con la parte de mostrar las actividades que el centro oferta, se pierden las opciones de inscripción, compra o favoritos pero de todas las aplicaciones estudiadas es la única que contiene información deportiva y cultural de eventos.



**Ilustración 12: Universidad de Alcalá de Henares.**

Tras el análisis de las universidades madrileñas, llegamos a la conclusión de que el ámbito deportivo y cultural no está recogido entre una de las principales necesidades para nuestra competencia y las universidades que más se aproximan a este entorno lo hacen de manera informativa y sin posibilidades reales de interactuar con el sistema haciendo reservas, inscripciones o compras.

Las expectativas a nivel nacional tampoco son mucho mejores, no se han estudiado la totalidad de las universidades pero si se ha hecho un barrido de las más importantes de España, siempre a nivel público y el resultado es similar. Está claro que se podría realizar un estudio más profundo y extenso sobre universidades privadas o universidades europeas o internacionales, pero a nivel práctico este estudio no tiene mucha relevancia debido a que es un gasto extra de recursos y nuestros principales competidores son las universidades públicas de Madrid.

Podríamos llegar a la conclusión de que la Universidad Carlos III de Madrid es un referente en cuanto al desarrollo de aplicaciones, tiene un catálogo de aplicaciones mayor que el de la mayoría de universidades y posee mayor diversidad de aplicaciones, como en el caso de Campus Life con el ámbito del deporte y la cultura.



## 2.6 TECNOLOGÍA UTILIZADA.

A continuación describiremos las tecnologías que se utilizarán para el desarrollo de nuestra aplicación, como se comentó en el anterior apartado, los sistemas en los que se podrá ejecutar Campus Life son Android e iOS, centrando este documento en el segundo de los dos sistemas operativos.

Como dispositivo de pruebas utilizaremos un iPhone 6 de 16 Gb de capacidad con 4,7 pulgadas de pantalla y un procesador con chip A8 y arquitectura de 64 bits. El sistema operativo será la nueva versión iOS 9 que viene instalada por defecto en nuestro iPhone.

Cabe destacar que los desarrollos para iOS tienen un gran control de calidad por parte de Apple y como nuestra aplicación dará soporte a dispositivos que tengan instalada hasta la versión 7.1 de iOS, también se realizaran pruebas en un iPhone 5 con sistema operativo iOS 8 y en un iPhone 4s con sistema operativo iOS 7.1, el resto de dispositivos de Apple serán analizados y testeados mediante simulador.

Una vez tenemos nuestros dispositivos de testeo analizaremos todas las herramientas básicas que utilizaremos para el desarrollo de la aplicación.



Ilustración 13: iPhone 6.

### 2.6.1 HERRAMIENTAS BÁSICAS.

Hay una serie de herramientas básicas para la realización de todo proyecto iOS, Apple como es conocido, es una empresa muy cerrada, fabrica sus propios dispositivos, crea sus propios sistemas y por lo general todo lo que diseña funciona a la perfección con sus iguales, pero nefastamente con los de la competencia.

Como no iba a ser diferente, para desarrollar una aplicación para iOS necesitaremos un ordenador de Apple, en nuestro caso utilizaremos un Mac Mini del 2012 con procesador Intel i5 y 8 Gb de memoria RAM. Si no disponemos de un ordenador de Apple nos será imposible realizar la aplicación pues el entorno de desarrollo es exclusivo de la empresa.

### 2.6.2 ENTORNO DE DESARROLLO: XCode.

XCode, es el entorno de desarrollo creado por Apple para desarrollar aplicaciones para sus dispositivos, únicamente se puede ejecutar en sistemas de Apple y es exclusivo de la empresa. Cabe destacar que además se debe de poseer una licencia de desarrollador que se empareja a nuestro dispositivo, además si queremos publicar nuestras aplicaciones debemos de tener una licencia de publicación que se asocia a un equipo de trabajo donde debe de estar incluido el desarrollador y por último, todos los dispositivos de prueba deben de estar también asociados al equipo de trabajo, al desarrollador que puede utilizarlo y al proyecto sobre el que se puede ejecutar.

Como se puede observar, está todo cuidado al más mínimo detalle y no hay tiempo para la improvisación, cada desarrollo en iOS o Mac OS está bien atado y en cada momento se sabe quien lo está desarrollando, a que equipo pertenece, a que empresa desarrolladora está ligado y en que dispositivos se ejecuta. Por si fuera poco, una vez mandamos nuestra aplicación terminada a Apple, esta sufre una batería de pruebas y estándares que debe de superar. y el más mínimo detalle, incluso estético, puede hacer que sea rechazada.



XCode 7 será la versión que utilizaremos para nuestro proyecto, en concreto la versión 7.3 (7D175).

Al igual que los dispositivos, Apple suele lanzar nuevas versiones tanto de sistema operativo como de entorno de desarrollo cada vez que

salen al mercado sus nuevos dispositivos móviles, evidentemente el entorno tiene que soportar las nuevas funcionalidades de los nuevos smartphones.

### 2.6.3 LENGUAJE DE DESARROLLO: OBJETIVE-C.

Como en todas las fases del proyecto, nos ha cogido por sorpresa una macro actualización de todo, dispositivos, sistema operativo, IDE y como no, también de lenguaje de programación. Hasta la fecha desde que saliera el primer XCode allá por el año 2003, el lenguaje de programación utilizado por Apple para el desarrollo de sus aplicaciones has sido siempre Objective-C un lenguaje de programación orientado a objetos, creado como un súper conjunto de C.

Pero con la actualización de XCode 6, vino la opción de comenzar a desarrollar aplicaciones con un nuevo lenguaje de programación (Swift), entonces ¿Porqué no utilizamos Swift?.

La respuesta es sencilla, adaptarse a un nuevo lenguaje de programación cuesta mucho tiempo y esfuerzo, y lanzar aplicaciones que pueden descargarse miles de personas, desarrolladas en un lenguaje que no se domina, puede generar el doble de errores en la aplicación y como consecuencia directa el desencanto de los usuarios con la aplicación.

Por esa razón el lenguaje de programación utilizado será Objective-C. Es robusto, tiene muchísima documentación y ejemplos, y actualmente es más seguro para programar.



Ilustración 14: Objective-C vs Swift.

#### 2.6.4 COCOA PODS.

Cocoa pods es un gestor de dependencias para librerías de Objective –C, los usuarios generan código que almacenan en la nube, en sitios web como pueda ser Git Hub en forma de librerías, con documentación para instalarlas en nuestros proyectos y disfrutar de sus funcionalidades. Está claro que cuanto más librerías utilicemos, más dependientes seremos en nuestro código pero más carga de trabajo liberaremos ganando en tiempo y diseño. Algunas de las librerías son muy útiles y están muy bien diseñadas, de forma que es incluso mejor recurrir a estas técnicas, que intentar desarrollarlo por nuestra cuenta.

Todo esto tiene un gran problema, las librerías son diseñadas para un sistema y unas especificaciones técnicas concretas, cuando el desarrollador de la librería decide actualizarla, nuestra antigua versión queda en desuso y puede reportar errores, dejar de funcionar o perder características, para ello nació Cocoa Pods, que para los que trabajan habitualmente con J2EE sería similar a Apache Maven.



Ilustración 15: Cocoa Pods.

Cocoa actualiza nuestras dependencias y hace que nuestro proyecto se desarrolle paralelamente a sus librerías de manera que siempre tenemos todos los detalles y funcionalidades previstas.

Para crear nuestra aplicación, hemos utilizado algunas librerías de Cocoa Pods y las hemos añadido como dependencias de nuestro proyecto, estas son algunas de ellas:

- **AFNetworking:** Es una librería que nos ayuda a realizar todo tipo de conexiones entre cliente y servidor, transito de datos, cacheo de imágenes, etc. <https://cocoapods.org/pods/AFNetworking>
- **iCarousel:** Es una librería que nos permite crear un carrusel de vistas, en nuestra aplicación es utilizada para mostrar las actividades de manera sencilla en una sola vista, el usuario puede deslizar el dedo y recorrer fácilmente las actividades ofertadas por un campus. <https://cocoapods.org/pods/iCarousel>
- **MMPProgressHUD:** Es una librería que diseña una pequeña ventana emergente con información, se utiliza mucho para indicar al usuario que la vista que está cargándose contiene elementos pesados que pueden retrasar la visibilidad y el tiempo que debe de esperar. Se utiliza mucho al

cargar las actividades y al iniciar la aplicación tras el logueo del usuario.

<https://cocoapods.org/pods/MMProgressHUD>

- **GoogleMaps:** Es una librería que nos ayuda a trabajar con los mapas de Google, Apple posee sus propios mapas y es más sencillo de trabajar con ellos, pero la universidad tiene proyectos avanzados con Google donde se muestran los edificios, sus plantas y el nombre interno de las aulas. Con toda esta potencia de mapas y las oportunidades que nos brindan, utilizar GoogleMaps en nuestro proyecto sale muy rentable a pesar de tener que depender de librerías externas. Se utiliza en el apartado de contáctanos. <https://cocoapods.org/pods/GoogleMaps>
- **HHPanningTableViewCell:** Es una librería que genera instancias de celdas especiales, estas celdas se pueden deslizar y tienen bajo de ellas funcionalidades especiales, se han utilizado en las listas de Actividades favoritas y actividades inscritas para poder visualizar, borrar o compartir de manera rápida y sencilla y sin necesidad de cambiar de vista y entrar en el detalle concreto de cada una de las actividades. <https://cocoapods.org/pods/HHPanningTableViewCell>
- **Fabric:** Es una librería que trabaja sobre la antigua Crashlytics, un gestor de actividad sobre errores y fallos que se producen sobre la aplicación, esta librería nos permite marcar puntos clave en el código y reportar información específica para la resolución de los errores. Es muy efectiva y nos permite controlar varios detalles como, dispositivo sobre el que se produce el error, batería y almacenamiento disponible durante el error, datos marcados de interés por el desarrollador como puede ser, vista en la que se ha producido el error. <https://cocoapods.org/pods/Fabric>
- **REFrostedViewController:** es una librería que genera de manera fácil y sencilla un menú lateral deslizable, accesible desde cualquier vista de la aplicación y que nos permite tener un menú completo con todas las funcionalidades de la aplicación sin malgastar espacio de pantalla dándonos mucho juego en el diseño de las vistas. <https://cocoapods.org/pods/REFrostedViewController>
- **AeroGear-Push:** es una librería que nos facilita la tarea a la hora de enviar y recibir notificaciones, actualmente no se utiliza, pero en versiones posteriores se pretende poder recibir notificaciones sobre las plazas que quedan en una actividad, las nuevas actividades ofertadas, los cobros sobre inscripciones, etc. <https://cocoapods.org/pods/AeroGear-Push>

- **Flurry:** Es una librería parecida a Fabric pero más enfocada al uso de estadísticas y recopilación de datos de uso, con Flurry podemos saber que vistas se visitan más que actividades tienen mejor acogida, o si merece la pena modificar la aplicación para lanzar una nueva versión modificando vistas o generando algunas nuevas funcionalidades.  
<https://cocoapods.org/pods/Flurry-iOS-SDK>

Estas son las más importantes librerías externas que hemos utilizado para nuestra aplicación, como se puede observar, tenemos funcionalidades inmediatas que costarían mucho tiempo de desarrollar y con resultados perfectos y muy profesionales.

Hay otras alternativas a Cocoa Pods como puede ser el uso de Carthage otro gestor de dependencias, ¿Por qué no utilizamos Carthage?, Carthage fue el primer gestor de dependencias en trabajar con Swift, de hecho, la propia Carthage está escrita en Swift.

Utiliza exclusivamente marcos dinámicos en lugar de bibliotecas estáticas, esta es la única manera de distribuir binarios Swift que son compatibles con iOS 8 en adelante, al estar nuestra aplicación programada en Objective-C este gestor de dependencias no nos ayuda en el desarrollo, por ello nos decantamos por Cocoa Pods.

#### 2.6.5 WEB SERVICES.

Se trata de una tecnología para comunicar aplicaciones entre sí utilizando estándares y protocolos específicos. Una de las principales ventajas de los servicios web es la posibilidad de realizar una comunicación entre aplicaciones diferentes, desarrolladas en lenguajes diferentes y que pueden ejecutarse sobre diferentes plataformas. Esta ventaja hace de los servicios web uno de los más utilizados en la actualidad para la comunicación entre aplicaciones.

Los servicios web tienen grandes ventajas como la interoperabilidad, el fomento de protocolos y estándares textuales para mejorar su comprensión y acceso a contenido y es capaz de combinar servicios para integrarlos independientemente de su localización y su compañía, sin embargo también posee algunos inconvenientes como pueden ser su bajo rendimiento con respecto a otros servicios de comunicación e intercambio de datos y su utilización del protocolo http implica la pérdida de seguridad al saltarse el firewall de los enrutadores.

Dentro de los servicios web hay una gran variedad de estilos arquitectónicos que utilizan el protocolo http para intercambiar datos entre aplicaciones pero

en este apartado únicamente vamos a mencionar los servicios REST que son los que se han utilizado para el desarrollo de Campus Life.



**Ilustración 16: RESful.**

REST es muchas veces llamado también RESTful se basa en http que posee una gran escalabilidad gracias a sus protocolos cliente-servidor sin estado, operaciones bien definidas como pueden ser POST, GET, PUT o DELETE, una sintaxis universal o el uso de hipermedios.

Con los servicios web REST podemos asignar un identificador a un objeto mediante una URL, este objeto es accesible mediante dicha URL y tiene una serie de operaciones que devuelven información del objeto, de esta manera mediante http podemos comunicar las aplicaciones sin que cada una sepa nada de la otra.

#### **2.6.6 INTELIJ IDEA.**

Se trata de un entorno de desarrollo integrado compuesto por un conjunto de herramientas de programación de código abierto multiplataforma, permite el desarrollo de aplicaciones java y es una de las posibilidades más óptimas para desarrollar servicios web junto con algunos Frameworks como Spring.



**Ilustración 17: IDE IntelliJ IDEA.**

La conocida versión Android Studio de Google, un código abierto IDE para aplicaciones de Android, está basada en el código abierto en la edición comunitaria de IntelliJ IDEA y actualmente se consolida como uno de los principales competidores de ECIPSE.



### 2.6.7 SPRING BOOT.

Se trata de una tecnología del Framework Spring que nos ayuda a automatizar las tareas de creación de un proyecto gestionar las dependencias y generación de los paquetes para realizar el despliegue en un servidor de aplicaciones web.



**Ilustración 18: Spring Boot.**

Con Spring Boot podemos generar de manera automática y rápida mediante unos sencillos comandos un nuevo proyecto con su jerarquía de carpetas y su estructura específica dependiendo del tipo de proyecto, en nuestro caso un Web Service para las llamadas entre los clientes de Campus Life y la conexión con nuestra base de datos donde almacenamos la información.

### 2.6.8 SERVIDOR WEB.

Se trata de uno de los puntos fuertes de nuestro proyecto y es donde se almacenarán y gestionarán las aplicaciones web, en nuestro caso almacenaremos y gestionaremos los paquetes de servicio web de Campus Life y con ellos se desplegará el servicio, si el servidor cae y su sistema se ve afectado los servicios web se desconectarán y nuestra aplicación dejará de funcionar correctamente debido a la falta de comunicación.



**Ilustración 19: Tomcat.**

Para este proyecto se ha utilizado el servidor web Tomcat en su última versión estable 8.0.30, Tomcat pertenece a la compañía Apache Software Foundation y se trata de un proyecto OpenSource por lo que no necesitaremos aplicar gastos de licencia, además es muy ligero y su consumo de recursos es bajo, es uno de los servidores más sencillos de utilizar y gestionar aunque pierde en cuanto a abanico de funcionalidades con respecto a otros servidores como puedan ser JBoss, es totalmente compatible con Java y multiplataforma lo cual lo convierte en el servidor perfecto para ejecutar nuestros servicios.



### 2.6.9 GESTOR DE BASE DE DATOS.

Se trata de otro de los puntos fuertes de nuestro proyecto y es donde se almacenarán y gestionarán los datos, nuestro SGBD tiene que ser fiable, los datos no se corrompen ni hay pérdida de los mismos, también tiene que ser seguro, debe de permitir el acceso únicamente a las aplicaciones o usuarios con permiso y debe de estar preparado ante cualquier amenaza externa que ponga en riesgo la integridad de los datos.

Para este proyecto se ha utilizado el SGBD MySQL en su última versión estable 5.7.15, se trata de un gestor de base de datos relacional y fue desarrollado por Sun Microsystems y vendida a Oracle a posteriori, se trata del SGBD más popular OpenSource con licencia GPL por lo que es totalmente gratuito, destaca por un rendimiento alto en la gestión de recursos y una gran capacidad, permite la concurrencia de datos y es multiplataforma, además su nivel de seguridad es alto.



Existen bases de datos más potentes y con más opciones pero no suelen ser de licencia libre y sus costes disparan el presupuesto, por todo ello se trata de un SGBD perfecto para la realización de nuestro proyecto.



## CAPÍTULO 3: ANÁLISIS DEL SISTEMA.

En el capítulo número 3 se describen una de las tres fases fundamentales para realizar un proyecto, el análisis, este punto es crítico pues de él dependen el resto de puntos y un error en esta fase desencadena múltiples errores en el desarrollo, debido a la falta de datos donde apoyarnos, es un proceso muy creativo y de consenso entre los departamentos de desarrollo y el cliente.

En esta fase se asientan las bases del proyecto y se analizan las posibilidades reales de implementar las ideas, se dividirá en 2 partes, el análisis de requisitos, el análisis de los casos de uso.

### 3.1 REQUISITOS DE USUARIO

**Identificador: <tipo de requisito> - <número>**

Nombre:	
Descripción:	
Necesidad:	
Prioridad:	
Fuente:	
Versión:	
Fecha:	

**Tabla 1 Descripción Tabla de Requisitos de Usuario**

Los campos se rellenarán en función de la siguiente definición:

- **Identificador:** Código que identifica de manera unívoca a cada requisito. Consta de dos partes: tipo de requisito y número.
- **Tipo de requisito:** Existen dos tipos de requisitos: requisitos de usuario (RU) y requisitos sistema (RS), en este apartado trataremos los requisitos de usuario y podrán ser de dos tipos funcionales de capacidad (RUC) o de restricción (RUR).
- **Número:** Al modificar el tipo de requisito, la numeración deberá empezar de nuevo por el 1.
- **Nombre:** Título que identifica el requisito.
- **Descripción:** Definición sobre el requisito.
- **Necesidad:** Grado en el que el requisito debe ser satisfecho. Puede tomar los valores de “Muy Alto”, “Alto”, “Medio”, “Bajo” y “Muy Bajo”.
- **Prioridad:** Nivel de urgencia con el que debe cumplirse el requisito. Puede tomar los valores de “Muy Alto”, “Alto”, “Medio”, “Bajo” y “Muy Bajo”.
- **Fuente:** Indica la procedencia del requisito, pudiendo proceder de documentos o personas.
- **Versión:** indica el número de versión del requisito.
- **Fecha:** indica la fecha en la que se hace público el requisito.

### 3.1.1 REQUISITOS DE CAPACIDAD

#### Identificador: RUC- 01

Nombre:	Identificación de Usuario.
Descripción:	El sistema debe de permitir al usuario identificarse para poder acceder a los servicios que proporciona la aplicación.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 2: Requerimiento de Usuario Capacidad 1.

#### Identificador: RUC - 02

Nombre:	Información de Acceso.
Descripción:	El sistema debe de alertar al usuario en caso de que surja algún problema en el proceso de identificación.
Necesidad:	Media.
Prioridad:	Baja.
Fuente:	Encargado de Seguridad, Analista.
Versión:	1.0.0
Fecha:	19/5/2015

Tabla 3: Requerimiento de Usuario Capacidad 2.

#### Identificador: RUC - 03

Nombre:	Raíz de servicios.
Descripción:	<p>El sistema debe de permitir al usuario acceder desde cualquier vista y en cualquier momento a las diferentes funcionalidades que ofrece la aplicación:</p> <ul style="list-style-type: none"><li>• Comprar tarjeta.</li><li>• Actividades por campus.</li><li>• Favoritas.</li><li>• Mis Actividades.</li><li>• Contacto.</li><li>• Promociones.</li></ul>

	<ul style="list-style-type: none"> <li>• Información.</li> <li>• Salir.</li> </ul>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 4: Requerimiento de Usuario Capacidad 3.**

**Identificador: RUC - 04**

Nombre:	Compartir.
Descripción:	El sistema debe de permitir al usuario compartir en RRSS los servicios prestados por la aplicación.
Necesidad:	Media.
Prioridad:	Baja.
Fuente:	Analista, ESES, Comunicación.
Versión:	1.0.0
Fecha:	19/5/2015

**Tabla 5: Requerimiento de Usuario Capacidad 4.**

**Identificador: RUC - 05**

Nombre:	Actividades por campus.
Descripción:	El sistema debe de permitir al usuario visualizar las actividades que oferta cada campus de la universidad.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 6: Requerimiento de Usuario Capacidad 5.**

**Identificador: RUC - 06**

Nombre:	Tipos de actividad.
---------	---------------------

Descripción:	El sistema debe de permitir al usuario cambiar el tipo de actividades que está visualizando.
Necesidad:	Media.
Prioridad:	Baja.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 7: Requerimiento de Usuario Capacidad 6.****Identificador: RUC - 07**

Nombre:	Información de actividad.
Descripción:	<p>El sistema debe de permitir al usuario visualizar la información de cada actividad, debe de mostrar:</p> <ul style="list-style-type: none"><li>• Nombre.</li><li>• Imagen.</li><li>• Descripción.</li><li>• Precio.</li></ul>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 8: Requerimiento de Usuario Capacidad 7.****Identificador: RUC - 08**

Nombre:	Actividad favorita.
Descripción:	El sistema debe de permitir al usuario marcar y desmarcar como favoritas las actividades.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 9: Requerimiento de Usuario Capacidad 8.**

**Identificador: RUC - 09**

Nombre:	Video de la actividad.
Descripción:	El sistema debe de permitir al usuario reproducir el video de las actividades, en caso de que exista.
Necesidad:	Media.
Prioridad:	Baja.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	19/5/2015

**Tabla 10: Requerimiento de Usuario Capacidad 9.**

**Identificador: RUC - 10**

Nombre:	Inscripción de la actividad.
Descripción:	El sistema debe de permitir al usuario inscribirse en las actividades.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 11: Requerimiento de Usuario Capacidad 10.**

**Identificador: RUC - 11**

Nombre:	Horario de la actividad.
Descripción:	El sistema debe de permitir al usuario ver el horario de las actividades.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 12: Requerimiento de Usuario Capacidad 11.**



**Identificador: RUC - 12**

Nombre:	Comprar tarjeta TE.
Descripción:	El sistema debe de permitir al usuario comprar las diversas tarjetas ofertadas por Espacio Estudiantes.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 13: Requerimiento de Usuario Capacidad 12.****Identificador: RUC - 13**

Nombre:	Añadir o modificar número de cuenta.
Descripción:	El sistema debe de permitir al usuario añadir o modificar un número de cuenta para comprar su tarjeta TE o inscribirse en una actividad.
Necesidad:	Media.
Prioridad:	Baja.
Fuente:	Analista, ESES, Comunicación.
Versión:	1.0.0
Fecha:	19/5/2015

**Tabla 14: Requerimiento de Usuario Capacidad 12.****Identificador: RUC – 14**

Nombre:	Actividades favoritas.
Descripción:	El sistema debe de permitir al usuario visualizar todas las actividades marcadas como favoritas.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 15: Requerimiento de Usuario Capacidad 14.**

### Identificador: RUC - 15

Nombre:	Borrar actividades favoritas.
Descripción:	El sistema debe de permitir al usuario borrar las actividades favoritas.
Necesidad:	Media.
Prioridad:	Media.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	19/5/2015

Tabla 16: Requerimiento de Usuario Capacidad 15.

### Identificador: RUC - 16

Nombre:	Ampliar información de actividad favorita.
Descripción:	<p>El sistema debe de permitir al usuario visualizar una actividad favorita, debe de mostrar:</p> <ul style="list-style-type: none"><li>• Nombre.</li><li>• Imagen.</li><li>• Descripción.</li><li>• Precios.</li><li>• Horario.</li></ul>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 17: Requerimiento de Usuario Capacidad 16.

### Identificador: RUC – 17

Nombre:	Actividades inscritas.
Descripción:	El sistema debe de permitir al usuario visualizar todas las actividades en las que el usuario se encuentra inscrito.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.

Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 18: Requerimiento de Usuario Capacidad 17.

**Identificador: RUC - 18**

Nombre:	Ampliar actividad inscrita.
Descripción:	<p>El sistema debe de permitir al usuario visualizar una actividad inscrita, debe de mostrar:</p> <ul style="list-style-type: none"><li>• Nombre.</li><li>• Imagen.</li><li>• Descripción.</li><li>• Precios.</li></ul>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 19: Requerimiento de Usuario Capacidad 18.

**Identificador: RUC – 19**

Nombre:	Promociones.
Descripción:	<p>El sistema debe de permitir al usuario visualizar las promociones ofertadas por Espacio Estudiantes, debe de mostrar:</p> <ul style="list-style-type: none"><li>• Nombre.</li><li>• Imagen.</li><li>• Descripción.</li><li>• Fecha fin de promoción.</li></ul>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0

Fecha: 4/3/2015

**Tabla 20: Requerimiento de Usuario Capacidad 19.**

**Identificador: RUC – 20**

Nombre:	Enlace externo.
Descripción:	El sistema debe de permitir al usuario acceder al enlace externo de la promoción para poder visualizarla completamente.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 21: Requerimiento de Usuario Capacidad 20.**

**Identificador: RUC – 21**

Nombre:	Tipos de promoción.
Descripción:	El sistema debe de permitir al usuario cambiar entre las promociones para universitarios y las promociones para personal con tarjeta TE.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 22: Requerimiento de Usuario Capacidad 21.**

**Identificador: RUC – 22**

Nombre:	Contacto.
Descripción:	<p>El sistema debe de permitir al usuario contactar con las oficinas de Espacio Estudiantes mediante los siguientes métodos:</p> <ul style="list-style-type: none"> <li>• Redes sociales.</li> <li>• Correo electrónico.</li> <li>• Llamada telefónica.</li> <li>• Localización de oficinas mediante mapas.</li> </ul>

Necesidad:	Media.
Prioridad:	Baja.
Fuente:	Analista, ESES, Comunicación.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 23: Requerimiento de Usuario Capacidad 22.****Identificador: RUC - 23**

Nombre:	Información.
Descripción:	El sistema debe de permitir al usuario visualizar información sobre la aplicación.
Necesidad:	Baja.
Prioridad:	Muy Baja.
Fuente:	Analista, ESES, Comunicación.
Versión:	1.0.0
Fecha:	19/5/2015

**Tabla 24: Requerimiento de Usuario Capacidad 23.****Identificador: RUC - 24**

Nombre:	Salir.
Descripción:	El sistema debe de permitir al usuario salir de la sesión iniciada.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 25: Requerimiento de Usuario Capacidad 24.**

### 3.1.2 REQUISITOS RESTRICCIÓN

#### Identificador: RUR - 01

Nombre:	Identificación.
Descripción:	El sistema no debe de permitir al usuario poder acceder a ninguna de las funcionalidades de la aplicación sin haber sido verificados con anterioridad sus credenciales de acceso.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 26: Requerimiento de Usuario Restricción 1.

#### Identificador: RUR - 02

Nombre:	Página principal.
Descripción:	La primera página que debe de observar el usuario al iniciar el sistema es las actividades del campus Getafe.
Necesidad:	Media.
Prioridad:	Media.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 27: Requerimiento de Usuario Restricción 2.

#### Identificador: RUR - 03

Nombre:	No visualizar actividades inscritas.
Descripción:	El sistema no debe de permitir al usuario poder visualizar en la página de actividades, ninguna actividad en la que se encuentre inscrito.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 28: Requerimiento de Usuario Restricción 3.

**Identificador: RUR - 04**

Nombre:	No realizar llamada.
Descripción:	El sistema no debe de permitir al usuario poder realizar llamadas a las oficinas de ESES, si no hay tarjeta SIM o no existe cobertura.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 29: Requerimiento de Usuario Restricción 4.

**Identificador: RUR - 05**

Nombre:	No bloquearse al cargar actividades.
Descripción:	El sistema no debe de bloquearse al cargar las actividades ni las promociones en caso de perder la conexión.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 30: Requerimiento de Usuario Restricción 5.

**Identificador: RUR - 06**

Nombre:	No permitir inscribirse sin número de cuenta.
Descripción:	El sistema no debe de permitir inscribirse en una actividad si no existe un registro de número de cuenta del usuario identificado en el sistema.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 31: Requerimiento de Usuario Restricción 6.

### Identificador: RUR - 07

Nombre:	No almacenar credenciales del usuario en el dispositivo.
Descripción:	El sistema no debe de almacenar ningún tipo de credenciales del usuario en el dispositivo.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 32: Requerimiento de Usuario Restricción 7.**

### Identificador: RUR - 08

Nombre:	Restricción de dispositivo y memoria.
Descripción:	El sistema no está optimizado para ejecutarse en móviles inferiores al iPhone 4S y necesita un mínimo de memoria para instalarse y poder almacenar datos.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 33: Requerimiento de Usuario Restricción 8.**



### 3.2 CASOS DE USO

Los casos de uso resumen las posibles interacciones que puede realizar el usuario en la aplicación, y se analiza el estado inicial en que se encuentra la aplicación, la acción realizada por el usuario, la respuesta producida por la aplicación y para concluir el estado final en el que quedará la aplicación tras la interacción con el usuario.

Para ello hemos diseñado una tabla, que recoge todos estos datos:

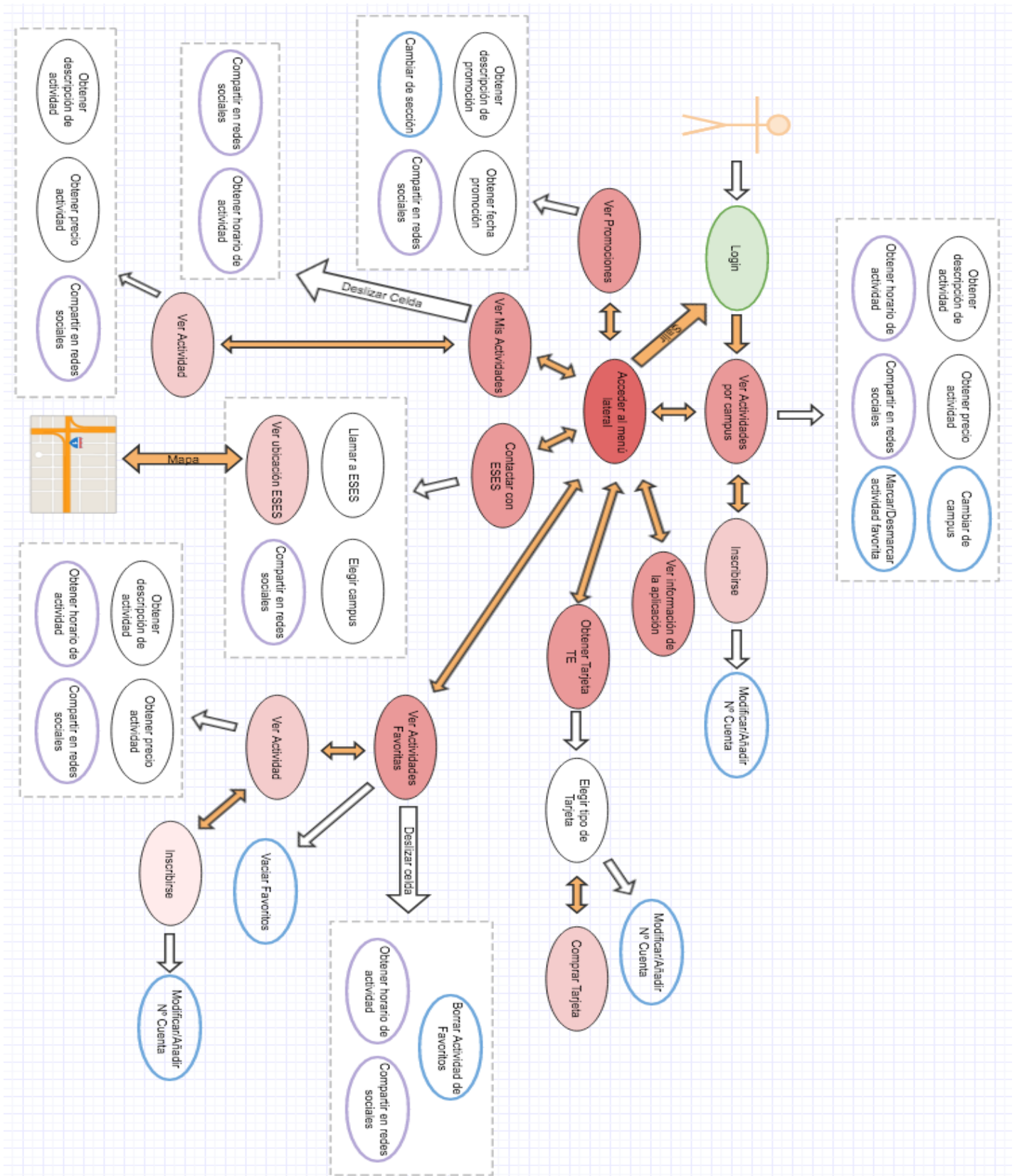
**Identificador: <tipo de caso de uso> - <número>**

Nombre:	
Actor:	
Objetivo:	
Descripción:	
Pre-condición:	
Pos-condición:	
Fuente:	

**Tabla 34 Descripción Tabla de Casos de Uso.**

Los campos se rellenarán en función de la siguiente definición:

- **Identificador:** Código que identifica de manera unívoca a cada uno de los casos de uso. Consta de dos partes: caso de uso (Fijo) y número.
- **Nombre:** Título que identifica cada caso de uso.
- **Actor:** Existen dos tipos de actores que son los encargados de desencadenar una interacción: usuario y sistema.
- **Objetivo:** Describe cual es la finalidad que se espera al producirse una interacción con la aplicación.
- **Descripción:** Definición detallada del caso de uso.
- **Pre-condición:** Estado inicial en el que se encuentra la aplicación antes de comenzar la interacción.
- **Pos-condición:** Estado final en el que se encuentra la aplicación después de terminar la interacción.
- **Fuente:** Requisito asociado al caso de uso.



El anterior diagrama, muestra de una sola vez, todos los casos de uso posibles en la aplicación.

Para poder comprenderlo mejor definiremos los componentes y su estructura:

- **Usuario:** se muestra mediante un muñeco.
- **Interacción:** se muestra mediante una elipse con una breve descripción de la interacción que puede realizar.
- **Transición:** se muestra mediante una flecha coloreada y marca el destino al que se dirige la aplicación al realizar una interacción, las transiciones pueden ser unidireccionales o bidireccionales.
- **Acciones:** se muestra mediante una flecha blanca y describe mediante un cuadro todas las interacciones posibles dentro de una vista.
- **Profundidad:** se mide mediante la intensidad de los colores de las elipses, cuanto más clara es una elipse más profunda es la navegación en la aplicación, el color verde denota la primera vista.
- **Localización:** se indica mediante un mapa e indica que en dicha vista se generara un mapa con la ubicación definida.
- **Pop-Ups emergentes:** se define mediante un borde lila e indica las interacciones que producen un pop-up emergente.
- **Refresco de vista:** se define mediante un borde azul e indica las interacciones que producen un refresco de información sobre la misma vista, sin transiciones.

En las siguientes tablas no se mostrarán las transiciones, pero se indicará el evento que las desencadena, así como algunos usos repetidos en diferentes vistas de la aplicación, que serán explicadas en una única tabla.

### Identificador: CU - 01

Nombre:	Login.
Actor:	Usuario.
Objetivo:	Acceder a las funcionalidades de la aplicación.
Descripción:	El usuario rellena los campos y estos son validados por el sistema.
Pre-condición:	Vista inicial de Login, campos vacíos.
Pos-condición:	Vista de Actividades de Getafe, credenciales correctas.
Fuente:	RUC – 01

Tabla 35: Caso de uso 1.

### Identificador: CU - 02

Nombre:	Alerta.
Actor:	Sistema.
Objetivo:	Verificar credenciales.
Descripción:	El sistema verifica los campos rellenos por el usuario y muestra un mensaje en caso de error.
Pre-condición:	Vista inicial de Login, campos rellenos, se pulsa el botón entrar.
Pos-condición:	Vista inicial de Login, campos rellenos, se muestra una alerta de error.
Fuente:	RUC - 02

Tabla 36: Caso de uso 2.

### Identificador: CU - 03

Nombre:	Menú.
Actor:	Usuario.
Objetivo:	Poder acceder a todas las funcionalidades de la aplicación desde un menú general. Se puede transitar a todas las vistas indicadas en el diagrama.
Descripción:	El usuario marca sobre Menú y accede a todas las funcionalidades principales de la aplicación.
Pre-condición:	Desde cualquier vista de la aplicación, acceder al Menú.
Pos-condición:	Vista del menú deslizante.
Fuente:	RUC - 03

Tabla 37: Caso de uso 3.

### Identificador: CU - 04

Nombre:	Visualizar Actividades.
Actor:	Usuario.
Objetivo:	Poder visualizar las actividades y su información.
Descripción:	El usuario cambia de actividad ofertada por campus avanzando o retrocediendo en la oferta.
Pre-condición:	Vista de Actividades del campus 'x', se hace scroll horizontal y desaparece la actividad 'x'.
Pos-condición:	Vista de Actividades del campus 'x', aparece la nueva actividad 'y'.
Fuente:	RUC - 07

Tabla 38: Caso de uso 4.

### Identificador: CU - 05

Nombre:	Cambio de Campus.
Actor:	Usuario.
Objetivo:	Refrescar Actividades del campus seleccionado.
Descripción:	El sistema cambia a la pestaña del campus seleccionado y refresca la vista general con las actividades ofertadas en dicho campus.
Pre-condición:	Vista de Actividades del campus 'x', se pulsa la pestaña del campus 'y'.
Pos-condición:	Vista de Actividades del campus 'y'.
Fuente:	RUC - 05

Tabla 39: Caso de uso 5.

### Identificador: CU - 06

Nombre:	Cambio de tipo de actividad.
Actor:	Usuario.
Objetivo:	Poder pasar rápidamente hacia una actividad perteneciente a otro tipo de actividades.
Descripción:	El usuario marca sobre un desplegable con los tipos de actividades que se ofertan en el campus 'x', que le traslada sin tener que pasar manualmente por todas las secciones al tipo de actividades deseado.
Pre-condición:	Vista de Actividades del campus 'x', se pulsa el desplegable y se selecciona un tipo de actividad.
Pos-condición:	Vista de Actividades del campus 'x', visualizando la primera actividad del tipo de actividades seleccionado.

Fuente: RUC - 06

**Tabla 40: Caso de uso 6.**

**Identificador: CU - 07**

Nombre:	Marcar Favorito.
Actor:	Usuario.
Objetivo:	Marcar una actividad como favorita para poder visualizarla después en la vista de Favoritas.
Descripción:	El usuario marca la opción de favorita sobre una actividad de un campus 'x'.
Pre-condición:	Vista de Actividades del campus 'x', opción desmarcada.
Pos-condición:	Vista de Actividades del campus 'x', opción marcada y añadida la actividad a la lista de Favoritas.
Fuente:	RUC - 08

**Tabla 41: Caso de uso 7.**

**Identificador: CU - 08**

Nombre:	Desmarcar Favorito.
Actor:	Usuario.
Objetivo:	Desmarcar una actividad como favorita para borrarla en la vista de Favoritas.
Descripción:	El usuario desmarca la opción de favorita sobre una actividad de un campus 'x'.
Pre-condición:	Vista de Actividades del campus 'x', opción marcada.
Pos-condición:	Vista de Actividades del campus 'x', opción desmarcada y borra la actividad a la lista de Favoritas.
Fuente:	RUC - 08

**Tabla 42: Caso de uso 8.**

**Identificador: CU - 09**

Nombre:	Vista de actividades en blanco.
Actor:	Sistema.
Objetivo:	No mostrar ninguna actividad por falta de conexión.
Descripción:	El sistema intenta cargar las actividades pero al recibir un time out del servidor carga la vista en blanco con un botón de recarga.
Pre-condición:	Vista de Actividades del campus 'x'.

Pos-condición:	Vista de Actividades del campus 'y' vacía con botón de recarga.
Fuente:	RUC – 07

**Tabla 43: Caso de uso 9.**

**Identificador: CU – 10**

Nombre:	Recargar vista de actividades.
Actor:	Usuario.
Objetivo:	Recargar la vista con las actividades del campus seleccionado.
Descripción:	El usuario marca la opción de recargar que hay en la vista vacía.
Pre-condición:	Vista de Actividades del campus 'x', vacía.
Pos-condición:	- Vista de Actividades del campus 'x', con todas las actividades ofertadas. - Vista de Actividades del campus 'x', vacía.
Fuente:	RUC – 07

**Tabla 44: Caso de uso 10.**

**Identificador: CU - 11**

Nombre:	Visualizar video.
Actor:	Usuario.
Objetivo:	Poder ver el video de la actividad, en caso de tener la opción de video.
Descripción:	El usuario marca sobre el botón de reproducir de la foto en caso de que la foto lo tenga, y podrá visualizar el video de la actividad.
Pre-condición:	Vista de Actividades del campus 'x', se pulsa sobre ver video de la actividad 'z'.
Pos-condición:	Vista reproductor de video, visualizando el video de la actividad 'z'.
Fuente:	RUC – 09

**Tabla 45: Caso de uso 11.**

**Identificador: CU – 12**

Nombre:	Scroll de la descripción.
Actor:	Usuario.
Objetivo:	Visualizar todo el contenido de la descripción.
Descripción:	El usuario hace scroll sobre la descripción para poder ver todo el contenido, en algunos dispositivos de pantalla más pequeña esta opción es muy habitual.

Precondición:	Vista de Actividades del campus 'x', descripción al inicio.
Poscondición:	Vista de Actividades del campus 'x', descripción deslizada.
Fuente:	RUC – 07

**Tabla 46: Caso de uso 12.**

**Identificador: CU – 13**

Nombre:	Inscripción.
Actor:	Usuario.
Objetivo:	Poder inscribirse en una actividad.
Descripción:	El usuario marca la opción de inscripción y se cambia de vista para realizar el proceso.
Pre-condición:	Vista de Actividades del campus 'x', se pulsa inscripción.
Pos-condición:	Vista de Inscripción con todos los detalles.
Fuente:	RUC – 10

**Tabla 47: Caso de uso 13.**

**Identificador: CU – 14**

Nombre:	Inscribirse.
Actor:	Usuario.
Objetivo:	Poder ver si se cumplen todos los requisitos para poder inscribirse e inscribirse en la actividad.
Descripción:	El usuario marca sobre inscribirse después de confirmar todos los requisitos para poder realizarlo.
Pre-condición:	Vista de inscripción de la actividad 'x', se pulsa sobre inscribirse en caso de que está opción esté habilitada.
Pos-condición:	- Vista de ventana emergente de confirmación para realizar inscripción, mostrando el nombre de la actividad y el precio de la misma. - Vista de inscripción de la actividad 'x', no se cumplen todos los requisitos y no está permitida la inscripción.
Fuente:	RUC - 10

**Tabla 48: Caso de uso 14.**

**Identificador: CU - 15**

Nombre:	Visualizar Horario.
Actor:	Usuario.



Objetivo:	Poder visualizar los horarios de una actividad.
Descripción:	El usuario marca sobre horarios y el sistema le muestra una ventana emergente con los horarios de la actividad.
Pre-condición:	Vista de Actividades del campus 'x', se pulsa sobre horarios.
Pos-condición:	Vista de Actividades del campus 'x', emerge una ventana con los horarios de la actividad.
Fuente:	RUC - 11

**Tabla 49: Caso de uso 15.**

### Identificador: CU - 16

Nombre:	Compartir en RRSS.
Actor:	Usuario.
Objetivo:	Poder compartir en RRSS la actividad que deseamos.
Descripción:	El usuario marca sobre compartir y puede compartir la información de la actividad en diferentes redes sociales.
Pre-condición:	Vista de Actividades del campus 'x', se pulsa sobre compartir.
Pos-condición:	Vista de Actividades del campus 'x', emerge una ventana con los las RRSS disponibles para compartir la actividad.
Fuente:	RUC - 04

**Tabla 50: Caso de uso 16.**

### Identificador: CU - 17

Nombre:	Favoritas.
Actor:	Usuario.
Objetivo:	Poder acceder a las actividades marcadas como favoritas.
Descripción:	El usuario marca en el Menú la opción Mis Favoritas y accede a una lista de actividades generada por el propio usuario con sus actividades preferidas.
Pre-condición:	Desde el menú se marca la opción de Mis Favoritas.
Pos-condición:	Vista de Favoritas.
Fuente:	RUC - 14

**Tabla 51: Caso de uso 17.**

### Identificador: CU - 18

Nombre:	Borrar Favorito.
---------	------------------

Actor:	Usuario.
Objetivo:	Poder borrar una actividad favorita.
Descripción:	El usuario marca en la vista Mis Favoritas sobre borrar y se eliminará la actividad favorita.
Pre-condición:	Desde la vista de Mis Favoritas, se desliza horizontalmente sobre la celda de la actividad que se desea borrar y aparecerá por debajo una serie de funciones, se marca sobre borrar.
Pos-condición:	Vista de Mis Favoritas, se recarga la lista y desaparece la actividad favorita, se elimina dicha actividad de la lista interna que almacena las actividades favoritas.
Fuente:	RUC - 15

**Tabla 52: Caso de uso 18.**

**Identificador: CU - 19**

Nombre:	Borrar Favoritas.
Actor:	Usuario.
Objetivo:	Poder borrar todas las actividades favoritas.
Descripción:	El usuario marca en la vista Mis Favoritas sobre borrar y se eliminarán la actividades favoritas.
Pre-condición:	Desde la vista de Mis Favoritas, se pulsa sobre borrar y se eliminarán todas las actividades favoritas.
Pos-condición:	Vista de Mis Favoritas, se recarga la lista y desaparecen las actividades favoritas, se eliminan las actividades de la lista interna que almacena las actividades favoritas.
Fuente:	RUC - 15

**Tabla 53: Caso de uso 19.**

**Identificador: CU - 20**

Nombre:	Compartir en RRSS actividad favorita.
Actor:	Usuario.
Objetivo:	Poder compartir en RRSS la actividad que deseamos.
Descripción:	El usuario desliza horizontalmente sobre la celda de la actividad que se desea compartir y aparecerá por debajo una serie de funciones, marca sobre compartir y puede compartir la información de la actividad en diferentes redes sociales.
Pre-condición:	Vista de Mis Favoritas con la celda deslizada, marca sobre compartir.
Pos-condición:	Vista de una ventana emergente con las posibles redes en las que

	compartir la información.
Fuente:	RUC - 04

**Tabla 54: Caso de uso 20.**

**Identificador: CU - 21**

Nombre:	Ver horario de actividad favorita.
Actor:	Usuario.
Objetivo:	Poder ver el horario de la actividad que deseamos.
Descripción:	El usuario desliza horizontalmente sobre la celda de la actividad que se desea compartir y aparecerá por debajo una serie de funciones, marca sobre horario y puede ver el horario de la actividad.
Pre-condición:	Vista de Mis Favoritas con la celda deslizada, marca sobre horario.
Pos-condición:	Vista de una ventana emergente con el horario de la actividad escogida.
Fuente:	RUC - 11

**Tabla 55: Caso de uso 21.**

**Identificador: CU - 22**

Nombre:	Ampliar información de actividad favorita.
Actor:	Usuario.
Objetivo:	Poder visualizar de manera más específica la información de una actividad favorita.
Descripción:	El usuario pulsa sobre la celda de la actividad favorita sobre la que se desea saber más información.
Pre-condición:	Vista de Mis Favoritas pulsar sobre la celda de la actividad que se desea ampliar.
Pos-condición:	Vista de Actividad favorita ampliada, con la información: <ul style="list-style-type: none"> <li>• Título.</li> <li>• Imagen o video.</li> <li>• Descripción.</li> <li>• Precios.</li> <li>• Horario.</li> <li>• Opción para poder inscribirse.</li> </ul>
Fuente:	RUC - 16

**Tabla 56: Caso de uso 22.**

### Identificador: CU - 23

Nombre:	Mis Actividades.
Actor:	Usuario.
Objetivo:	Poder acceder a las actividades inscritas.
Descripción:	El usuario marca en el Menú la opción Mis Actividades y accede a una lista de actividades en las que se encuentra inscrito.
Pre-condición:	Desde el menú se marca la opción de Mis Actividades.
Pos-condición:	Vista de Mis Actividades.
Fuente:	RUC - 17

Tabla 57: Caso de uso 23.

### Identificador: CU - 24

Nombre:	Compartir en RRSS actividad inscrita.
Actor:	Usuario.
Objetivo:	Poder compartir en RRSS la actividad que deseamos.
Descripción:	El usuario desliza horizontalmente sobre la celda de la actividad que se desea compartir y aparecerá por debajo una serie de funciones, marca sobre compartir y puede compartir la información de la actividad en diferentes redes sociales.
Pre-condición:	Vista de Mis Actividades con la celda deslizada, marca sobre compartir.
Pos-condición:	Vista de una ventana emergente con las posibles redes en las que compartir la información.
Fuente:	RUC - 04

Tabla 58: Caso de uso 24.

### Identificador: CU - 25

Nombre:	Ver horario de actividad inscrita.
Actor:	Usuario.
Objetivo:	Poder ver el horario de la actividad que deseamos.
Descripción:	El usuario desliza horizontalmente sobre la celda de la actividad que se desea compartir y aparecerá por debajo una serie de funciones, marca sobre horario y puede ver el horario de la actividad.
Pre-condición:	Vista de Mis Actividades con la celda deslizada, marca sobre horario.
Pos-condición:	Vista de una ventana emergente con el horario de la actividad escogida.

Fuente:	RUC - 11
---------	----------

**Tabla 59: Caso de uso 25.**

**Identificador: CU - 26**

Nombre:	Ampliar información de actividad inscrita.
Actor:	Usuario.
Objetivo:	Poder visualizar de manera más específica la información de una actividad inscrita.
Descripción:	El usuario pulsa sobre la celda de la actividad inscrita sobre la que se desea saber más información.
Pre-condición:	Vista de Mis Actividades pulsar sobre la celda de la actividad que se desea ampliar.
Pos-condición:	Vista de Actividad inscrita ampliada, con la información: <ul style="list-style-type: none"> <li>• Título.</li> <li>• Imagen o video.</li> <li>• Descripción.</li> <li>• Precios.</li> <li>• Horario.</li> </ul>
Fuente:	RUC - 16

**Tabla 60: Caso de uso 26.**

**Identificador: CU - 27**

Nombre:	Contacta.
Actor:	Usuario.
Objetivo:	Poder contactar con el departamento de ESES de diferentes maneras.
Descripción:	El usuario marca en el Menú la opción Contacta y se transita a una vista con diferentes opciones de contacto.
Pre-condición:	Desde el menú se marca la opción de Contacta.
Pos-condición:	Vista de Contacta, con las diferentes opciones de contacto: <ul style="list-style-type: none"> <li>• RRSS y correo.</li> <li>• Teléfono.</li> <li>• Mapa.</li> </ul>
Fuente:	RUC - 22

**Tabla 61: Caso de uso 27.**

### Identificador: CU - 28

Nombre:	Llamada.
Actor:	Usuario.
Objetivo:	Poder contactar vía telefónica con una oficina de ESES.
Descripción:	El usuario marca el campus al que desea llamar y pulsa sobre llamar.
Pre-condición:	El usuario marca el campus al que desea llamar y pulsa sobre llamar.
Pos-condición:	Se realiza una llamada al campus seleccionado.
Fuente:	RUC - 22

Tabla 62: Caso de uso 28.

### Identificador: CU - 29

Nombre:	Mapa.
Actor:	Usuario.
Objetivo:	Poder localizar la ubicación de las oficinas de ESES.
Descripción:	El usuario marca el campus al que desea localizar y pulsa sobre mapa.
Pre-condición:	El usuario marca el campus al que desea localizar y pulsa sobre mapa.
Pos-condición:	Se visualiza un mapa con la ubicación del campus seleccionado.
Fuente:	RUC - 22

Tabla 63: Caso de uso 29.

### Identificador: CU - 30

Nombre:	Promociones.
Actor:	Usuario.
Objetivo:	Poder ver la lista de promociones ofertadas con la tarjeta TE.
Descripción:	El usuario marca en el Menú la opción Promociones.
Pre-condición:	Desde el menú se marca la opción de Promociones.
Pos-condición:	Vista de Promociones.
Fuente:	RUC - 19

Tabla 64: Caso de uso 30.

**Identificador: CU - 31**

Nombre:	Información de la aplicación.
Actor:	Usuario.
Objetivo:	Poder acceder a información de la aplicación como el número de versión o quien la ha desarrollado.
Descripción:	El usuario marca en el Menú la opción información para visualizarla.
Pre-condición:	Desde el menú se marca la última opción.
Pos-condición:	Vista de información.
Fuente:	RUC - 23

**Tabla 65: Caso de uso 31.****Identificador: CU - 32**

Nombre:	Salir.
Actor:	Usuario.
Objetivo:	Poder salir de la aplicación con las credenciales logueadas.
Descripción:	El usuario marca en el Menú la opción salir.
Pre-condición:	Desde el menú se marca la opción de salir.
Pos-condición:	Vista de Login, se elimina el registro del token.
Fuente:	RUC - 24

**Tabla 66: Caso de uso 32.**

### 3.3 REQUISITOS DE SISTEMA.

**Identificador: <tipo de requisito> - <número>**

Nombre:	
Descripción:	
Necesidad:	
Prioridad:	
Fuente:	
Versión:	
Fecha:	

**Tabla 67: Descripción Tabla de Requisitos de Sistema.**

Los campos se rellenarán en función de la siguiente definición:

- **Identificador:** Código que identifica de manera unívoca a cada requisito. Consta de dos partes: tipo de requisito y número.
- **Tipo de requisito:** Existen dos tipos de requisitos: requisitos de usuario (RU) y requisitos software (RS), en este apartado trataremos los requisitos de software y podrán ser de 2 tipos, requisitos de sistema funcionales (RSF) y requisitos de sistema no funcionales (RSNF).
- **Número:** Al modificar el tipo de requisito, la numeración deberá empezar de nuevo por el 1.
- **Nombre:** Título que identifica el requisito.
- **Descripción:** Definición sobre el requisito.
- **Prioridad:** Nivel de urgencia con el que debe cumplirse el requisito. Puede tomar los valores de “Muy Alto”, “Alto”, “Medio”, “Bajo” y “Muy Bajo”.
- **Fuente:** Indica la procedencia del requisito, pudiendo proceder de documentos o personas.
- **Versión:** indica el número de versión del requisito.
- **Fecha:** indica la fecha en la que se hace público el requisito.



### 3.3.1 REQUISITOS FUNCIONALES.

#### Identificador: RSF - 01

Nombre:	Token de identificación.
Descripción:	La aplicación no almacenará ningún dato sensible del usuario, con sus credenciales se generará un token único que se utilizará para realizar las conexiones con el servidor. Al finalizar la sesión el token se destruirá, el token tienen un tiempo de vida útil, si se caduca se deberá pedir uno nuevo para poder realizar peticiones al servidor.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 68: Requerimiento de Sistema Funcional 1.**

#### Identificador: RSF - 02

Nombre:	Desplazamiento campos de texto.
Descripción:	Al abrir el teclado gran parte de la pantalla queda oculta, hay que desplazar los campos cuando el teclado emerge para evitar que puedan quedar debajo de él funcionalidades que nunca se puedan utilizar.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 69: Requerimiento de Sistema Funcional 2.**

#### Identificador: RSF - 03

Nombre:	Contraseña oculta.
Descripción:	La contraseña introducida por el usuario deberá de estar oculta de modo que se puedan ver la cantidad de caracteres que se introduce pero todos ellos serán sustituidos por puntos para mantener la confidencialidad de la misma.

Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 70: Requerimiento de Sistema Funcional 3.**

**Identificador: RSF - 04**

Nombre:	Alerta de identificación.
Descripción:	<p>El aviso de identificación se realizará mediante una parpadeo que mostrará los siguientes avisos dependiendo del error:</p> <ul style="list-style-type: none"> <li>• Aviso de usuario: Falta el usuario.</li> <li>• Aviso de contraseña: Falta la contraseña.</li> <li>• Aviso de identificación no válido: El usuario o la contraseña son erróneos, vuelva a introducirlos.</li> <li>• Aviso de identificación válido: Identificación correcta (Este último aviso se mostrará en color negro).</li> </ul>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 71: Requerimiento de Sistema Funcional 4.**

**Identificador: RSF - 05**

Nombre:	Menú Lateral.
Descripción:	<p>La raíz de servicios será un menú lateral deslizante que surgirá deslizando el dedo sobre la pantalla desde el lateral izquierdo hacia el lateral derecho, se verá parte de la vista por debajo del menú para dar la sensación de estar en superposición.</p> <p>Cada celda transitará a una vista específica en el siguiente orden según su diseño:</p> <ul style="list-style-type: none"> <li>• Vista Comprar tarjeta.</li> <li>• Vista Actividades por campus.</li> </ul>

	<ul style="list-style-type: none"> <li>• Vista Favoritas.</li> <li>• Vista Mis Actividades.</li> <li>• Vista Contacto.</li> <li>• Vista Promociones.</li> <li>• Vista Login.</li> <li>• Vista Información.</li> </ul> <p>También se puede acceder al menú pulsando el icono de la aplicación en cada vista, el botón salir elimina los datos almacenados de sesión del usuario y transita a la vista de Login.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 72: Requerimiento de Sistema Funcional 5.**

### Identificador: RSF - 06

Nombre:	Compartir.
Descripción:	Al pulsar el botón surge un menú emergente que contiene una lista de tipos de compartición, al pulsar sobre una de las opciones, se intenta abrir el cliente de aplicación , si no está instalado en el dispositivo se abrirá el navegador con la web solicitada.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 73: Requerimiento de Sistema Funcional 6.**

### Identificador: RSF - 07

Nombre:	Tipos de actividad.
Descripción:	Al pulsar el botón surge un menú emergente que contiene una lista de tipos de actividad, al pulsar sobre una de las opciones, el carrusel se desplazará de manera inmediata a la primera actividad perteneciente a la lista de actividades de dicho tipo, cerrando el menú.

	Cuando se desplaza manualmente por el carrusel y se cambia de tipo actividad el botón cambia su título por el nuevo tipo de actividad que está apareciendo en el carrusel.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 74: Requerimiento de Sistema Funcional 7.**

### Identificador: RSF - 08

Nombre:	Pestañas de Campus.
Descripción:	<p>Habrà 4 pestañas referentes a cada campus de la UC3M, siempre que se accede a la vista Actividades el primer campus en cargar será Getafe, la carga de actividades se produce una vez y se almacena en cache, al salir de la vista actividades se perderà la cache.</p> <p>Si no se cargan las actividades, se mostrarà el botón de recarga para poder volver a solicitarlas.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 75: Requerimiento de Sistema Funcional 8.**

### Identificador: RSF - 09

Nombre:	Carrusel.
Descripción:	<p>El carrusel tendrá principio y fin, no será circular.</p> <ul style="list-style-type: none"> <li>El nombre de la actividad será ajustable al ancho de pantalla dependiendo de la longitud del nombre, se reajustará el tamaño de la letra.</li> <li>La estrella de favorita al ser pulsada almacenará el id de la actividad y el usuario en un vector para mostrarlo después en la vista favoritas.</li> <li>La imagen se ajusta al ancho de pantalla y mantiene su</li> </ul>

	<p>proporción, si la imagen no se carga se mostrará una imagen por defecto, si la imagen no existe, se cargará otra imagen por defecto.</p> <ul style="list-style-type: none"> <li>• El video se mostrará mediante un botón de play y cargará el reproductor del sistema, cuando termina su ejecución retornará a la aplicación.</li> <li>• La descripción tendrá scroll y será de tamaño fijo, no es editable.</li> <li>• El precio en caso de no existir será por defecto 0€ para mostrar que es gratuita.</li> <li>• El botón de inscripción transita a la vista de inscripción y pasa como parámetro el id de la actividad.</li> <li>• El botón de horario abre una ventana emergente que muestra los días y el horario en que se imparte la actividad.</li> </ul>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 76: Requerimiento de Sistema Funcional 9.**

### Identificador: RSF - 10

Nombre:	Inscripción.
Descripción:	<p>La vista de inscripción constará de un botón inscripción que estará desactivado y únicamente se activará si se cumplen todos los requisitos de la lista.</p> <p>El único elemento que podemos modificar de los requisitos es el número de cuenta, que contiene 5 campos.</p> <ul style="list-style-type: none"> <li>• IBAN, Permite introducir números y letras y no más de 4 dígitos.</li> <li>• Oficina, Permite introducir números y no más de 4 dígitos.</li> <li>• Entidad, Permite introducir números y no más de 4 dígitos.</li> <li>• D.C, Permite introducir números y no más de 2 dígitos.</li> <li>• Número de Cuenta, Permite introducir números y no más de 10 dígitos.</li> </ul> <p>El botón añadir cambia automáticamente a modificar, en caso de empezar a introducir un nuevo número de cuenta.</p>

	Si se pulsa inscribirse, aparecerá un mensaje de alerta para confirmar de manera definitiva la inscripción.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 77: Requerimiento de Sistema Funcional 10.**

### Identificador: RSF - 11

Nombre:	Comprar TE.
Descripción:	Carga los tipos de tarjeta que puede comprar el usuario, dependiendo del tipo de tarjeta pulsado se transita a un tipo de compra específica.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 78: Requerimiento de Sistema Funcional 11.**

### Identificador: RSF - 12

Nombre:	Comprar TE confirmación.
Descripción:	<p>La vista de comprar constará de un botón comprar que estará desactivado y únicamente se activará si se cumple el requisitos número de cuenta que contiene 5 campos.</p> <ul style="list-style-type: none"> <li>• IBAN, Permite introducir números y letras y no más de 4 dígitos.</li> <li>• Oficina, Permite introducir números y no más de 4 dígitos.</li> <li>• Entidad, Permite introducir números y no más de 4 dígitos.</li> <li>• D.C, Permite introducir números y no más de 2 dígitos.</li> <li>• Número de Cuenta, Permite introducir números y no más de 10 dígitos.</li> </ul> <p>El botón añadir cambia automáticamente a modificar, en caso de empezar a introducir un nuevo número de cuenta.</p> <p>Si se pulsa comprar, aparecerá un mensaje de alerta para confirmar de</p>

	manera definitiva la compra.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 79: Requerimiento de Sistema Funcional 12.**

### Identificador: RSF - 13

Nombre:	Favoritos.
Descripción:	<p>La lista de favoritos es modificable, si pulsamos el botón borrar se borrarán todos los favoritos y la lista quedará vacía.</p> <p>Cada celda de actividad favorita es deslizante y contiene bajo de ella 3 opciones, borrar la actividad, que la elimina de la lista, compartir y ver horario.</p> <p>Al pulsar sobre una actividad favorita transitamos a otra vista que amplía la información de la misma.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 80: Requerimiento de Sistema Funcional 13.**

### Identificador: RSF - 14

Nombre:	Información Favoritos.
Descripción:	<p>El nombre de la actividad será ajustable al ancho de pantalla dependiendo de la longitud del nombre, se reajustará el tamaño de la letra.</p> <p>La imagen se ajusta al ancho de pantalla y mantiene su proporción, si la imagen no se carga se mostrará una imagen por defecto, si la imagen no existe, se cargará otra imagen por defecto.</p> <p>El video se mostrará mediante un botón de play y cargará el</p>

	<p>reproductor del sistema, cuando termina su ejecución retornará a la aplicación.</p> <p>La descripción tendrá scroll y será de tamaño fijo, no es editable.</p> <p>El precio en caso de no existir será por defecto 0€ para mostrar que es gratuita.</p> <p>El botón de inscripción transita a la vista de inscripción y pasa como parámetro el id de la actividad.</p> <p>El botón de horario abre una ventana emergente que muestra los días y el horario en que se imparte la actividad.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 81: Requerimiento de Sistema Funcional 14.**

### Identificador: RSF - 15

Nombre:	Mis Actividades.
Descripción:	<p>La lista de actividades inscrita es fija.</p> <p>Cada celda de actividad inscrita es deslizante y contiene bajo de ella 2 opciones, compartir y ver horario.</p> <p>Al pulsar sobre una actividad inscrita transitamos a otra vista que amplía la información de la misma.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 82: Requerimiento de Sistema Funcional 15.**



### Identificador: RSF - 16

Nombre:	Información inscritas.
Descripción:	<p>El nombre de la actividad será ajustable al ancho de pantalla dependiendo de la longitud del nombre, se reajustará el tamaño de la letra.</p> <p>La imagen se ajusta al ancho de pantalla y mantiene su proporción, si la imagen no se carga se mostrará una imagen por defecto, si la imagen no existe, se cargará otra imagen por defecto.</p> <p>El video se mostrará mediante un botón de play y cargará el reproductor del sistema, cuando termina su ejecución retornará a la aplicación.</p> <p>La descripción tendrá scroll y será de tamaño fijo, no es editable.</p> <p>El precio en caso de no existir será por defecto 0€ para mostrar que es gratuita.</p> <p>El botón de horario abre una ventana emergente que muestra los días y el horario en que se imparte la actividad.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 83: Requerimiento de Sistema Funcional 16.**

### Identificador: RSF - 17

Nombre:	Pestañas de Promociones.
Descripción:	<p>Habrà 2 pestañas referentes a los tipos de promociones ofertadas, siempre que se accede a la vista Promociones el primer tipo en cargar será Universitarios, la carga de promociones se produce una vez y se almacena en cache, al salir de la vista actividades se perderá la cache.</p> <p>Si no se cargan las promociones, se mostrará el botón de recarga para poder volver a solicitarlas.</p>
Necesidad:	Muy Alta.

Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 84: Requerimiento de Sistema Funcional 17.**

**Identificador: RSF - 18**

Nombre:	Carrusel promociones.
Descripción:	<p>El carrusel tendrá principio y fin, no será circular.</p> <ul style="list-style-type: none"> <li>• El nombre de la promoción será ajustable al ancho de pantalla dependiendo de la longitud del nombre, se reajustará el tamaño de la letra.</li> <li>• La imagen se ajusta al ancho de pantalla y mantiene su proporción, si la imagen no se carga se mostrará una imagen por defecto, si la imagen no existe, se cargará otra imagen por defecto.</li> <li>• La descripción tendrá scroll y será de tamaño fijo, no es editable.</li> <li>• La fecha de fin será por defecto temporada en caso de no existir.</li> <li>• El enlace externo, redirige en el navegador a la web que promociona la oferta.</li> </ul> <p>Las promociones que estén caducadas, no aparecerán en el carrusel.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 85: Requerimiento de Sistema Funcional 18.**

**Identificador: RSF - 19**

Nombre:	Contacto.
Descripción:	<p>Los botones superiores permiten el contacto con Twitter mediante mención, con Facebook mediante mensaje en muro y mediante correo electrónico, cada acción intentará abrir el cliente instalado en el dispositivo, en caso de no estar instalado se abrirá el navegador con las paginas correspondientes.</p>

	<p>La ruleta no es circular y tiene principio y fin.</p> <p>Los botones inferiores, son dependientes del valor de la ruleta y marcarán el número de teléfono del campus indicado o transitarán a la vista de mapa con la localización de la oficina indicada.</p>
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 86: Requerimiento de Sistema Funcional 19.**

### Identificador: RSF - 20

Nombre:	Hud.
Descripción:	<p>Todas las vistas que tienen componentes de carga de datos utilizarán un Hud que indicará al usuario el proceso de carga de datos, para que no tenga la sensación de bloqueo.</p> <p>El Hud surgirá como un comentario de alerta con masajes de carga, la vista de fondo perderá nitidez y se coloreará de un tono anaranjado.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 87: Requerimiento de Sistema Funcional 20.**

### 3.3.2 REQUISITOS NO FUNCIONALES.

#### Identificador: RSNF - 01

Nombre:	Vista de Login.
Descripción:	<p>La vista de Login constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Logotipo de la aplicación.</li> <li>• Usuario y campo para rellenar.</li> <li>• Contraseña y campo para rellenar.</li> <li>• Aviso de identificación.</li> <li>• Logotipo corporativo de la UC3M.</li> </ul> <p>El aviso de identificación será una etiqueta de color rojo.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

Tabla 88: Requerimiento de Sistema No Funcional 1.

#### Identificador: RSNF - 02

Nombre:	Vista de Menú.
Descripción:	<p>La vista de Menú constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Logotipo de la aplicación.</li> <li>• Nombre de usuario.</li> <li>• Comprar TE.</li> <li>• Actividades.</li> <li>• Favoritas.</li> <li>• Mis Actividades.</li> <li>• Promociones.</li> <li>• Contacto.</li> <li>• Salir</li> </ul>

	<ul style="list-style-type: none"> <li>Información (se sitúa en la esquina inferior derecha, será un icono de una letra i minúscula en el interior de un círculo, todo ello en color azul y fondo transparente).</li> </ul>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 89: Requerimiento de Sistema No Funcional 2.**

### Identificador: RSNF - 03

Nombre:	Vista de Actividad.
Descripción:	<p>La vista de Actividad constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>Menú, Nombre de vista, Compartir.</li> <li>Tipos de actividad.</li> <li>Carrusel.</li> <li>Pestañas de campus.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p> <p>Tipos de actividad se trata de un botón con detalle textual que muestra el tipo de actividad que se está visualizando en el carrusel, a la derecha del botón aparecerá un icono de una flecha apuntando hacia la parte inferior que denotará el efecto de lista de opciones.</p> <p>Las pestañas tendrán en su interior la letra inicial del campus al que hacen referencia, en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0

Fecha: 4/3/2015

**Tabla 90: Requerimiento de Sistema No Funcional 3.**

**Identificador: RSNF - 04**

Nombre:	Vista de Carrusel.
Descripción:	<p>La vista de Carrusel constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Nombre de actividad, Estrella favorita.</li> <li>• Imagen, video.</li> <li>• Descripción.</li> <li>• Precio actividad.</li> <li>• Inscripción, horario.</li> </ul> <p>La estrella de favoritos será de color naranja al estar seleccionada y transparente con borde naranja al estar deseleccionada.</p> <p>El botón de video será redondo de color rojo y con el icono de play en el interior en blanco.</p> <p>El botón de inscripción y horario estarán a la misma altura y tendrán texto identificativo en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 91: Requerimiento de Sistema No Funcional 4.**

**Identificador: RSNF - 05**

Nombre:	Vista de Inscripción confirmación.
Descripción:	<p>La vista de Inscripción confirmación constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre vista, Compartir.</li> <li>• Icono requisito, Usuario registrado.</li> </ul>

	<ul style="list-style-type: none"> <li>• Icono requisito, Perfil compatible.</li> <li>• Icono requisito, Plazas libres.</li> <li>• Icono requisito, Inscripción abierta.</li> <li>• Icono requisito, Número de cuenta, Botón añadir.</li> <li>• Precio tarjeta, Botón de compra.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p> <p>Los iconos pueden ser circulares de color rojo con un aspa en su interior para representar el no cumplimiento del requisito, o de color verde con un check en su interior que representa el cumplimiento del requisito.</p> <p>Los botones de añadir y compra serán en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 92: Requerimiento de Sistema No Funcional 5.**

### Identificador: RSNF - 06

Nombre:	Vista de Comprar TE.
Descripción:	<p>La vista de Comprar TE constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre vista, Compartir.</li> <li>• Imagen, descripción.</li> <li>• Tarjetas disponibles.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p>
Necesidad:	Alta.

Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 93: Requerimiento de Sistema No Funcional 6.**

**Identificador: RSNF - 07**

Nombre:	Vista de Comprar TE confirmación.
Descripción:	<p>La vista de Comprar TE confirmación constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre vista, Compartir.</li> <li>• Icono requisito, Número de cuenta, Botón añadir.</li> <li>• Precio tarjeta, Botón de compra.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 94: Requerimiento de Sistema No Funcional 7.**

**Identificador: RSNF - 08**

Nombre:	Vista de Favoritos.
Descripción:	<p>La vista de favoritos constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre vista, Borrar.</li> <li>• Lista de actividades favoritas.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p>



	El botón de borrar será el de por defecto en los sistemas iOS y en color naranja.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 95: Requerimiento de Sistema No Funcional 8.**

**Identificador: RSNF - 09**

Nombre:	Vista de Información favoritos.
Descripción:	<p>La vista de Información favoritos constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre de vista, Compartir.</li> <li>• Imagen, video.</li> <li>• Descripción.</li> <li>• Precio actividad.</li> <li>• Inscripción, horario.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 96: Requerimiento de Sistema No Funcional 9.**

**Identificador: RSNF - 10**

Nombre:	Vista de Mis actividades.
Descripción:	La vista de Mis actividades constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el

	<p>siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre vista, Compartir.</li> <li>• Lista de actividades inscritas.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 97: Requerimiento de Sistema No Funcional 10.**

### Identificador: RSNF - 11

Nombre:	Vista de Información inscritas.
Descripción:	<p>La vista de Información inscritas constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre de vista, Compartir.</li> <li>• Imagen, video.</li> <li>• Descripción.</li> <li>• Precio actividad.</li> <li>• Horario.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 98: Requerimiento de Sistema No Funcional 11.**

### Identificador: RSNF - 12

Nombre:	Vista de Promociones.
Descripción:	<p>La vista de Información inscritas constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre de vista, Compartir.</li> <li>• Carrusel.</li> <li>• Pestañas de perfil de usuario.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p> <p>La pestañas serán Universitarios para los usuarios estándar y ISCI/ITIC para los usuarios que tienen tarjeta TE.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 99: Requerimiento de Sistema No Funcional 12.**

### Identificador: RSNF - 13

Nombre:	Vista de Carrusel promoción.
Descripción:	<p>La vista de Carrusel promoción constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Nombre de promoción, Estrella favorita.</li> <li>• Imagen.</li> <li>• Descripción.</li> <li>• Fecha fin de promoción.</li> <li>• Enlace externo.</li> </ul>
Necesidad:	Alta.
Prioridad:	Alta.

Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 100: Requerimiento de Sistema No Funcional 13.**

**Identificador: RSNF - 14**

Nombre:	Vista de Contacto.
Descripción:	<p>La vista de Contacto constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p> <ul style="list-style-type: none"> <li>• Menú, Nombre de vista, Compartir.</li> <li>• Icono Twitter, Icono Facebook, Icono Mail.</li> <li>• Ruleta.</li> <li>• Icono Teléfono, Icono Mapa.</li> </ul> <p>El botón de menú será la imagen de la aplicación.</p> <p>El botón de compartir será el de por defecto en los sistemas iOS y en color naranja.</p> <p>Todos los botones serán circulares con el borde y el icono interior naranja, la ruleta contendrá los tres campus con oficinas de ESES, Getafe, Leganés y Colmenarejo.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 101: Requerimiento de Sistema No Funcional 14.**

**Identificador: RSNF - 15**

Nombre:	Vista de Información.
Descripción:	<p>La vista de Información constará de varios elementos que se organizarán de arriba a abajo y de izquierda a derecha según el siguiente orden de posicionamiento:</p>

	<ul style="list-style-type: none"> <li>• Icono UC3M.</li> <li>• Lista de departamentos.</li> <li>• Número de versión de aplicación.</li> </ul> <p>La lista de departamentos usará el color azul corporativo.</p>
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista, ESES.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 102: Requerimiento de Sistema No Funcional 15.**

### Identificador: RSNF - 16

Nombre:	Sistemas 7.0 de iOS.
Descripción:	El sistema mínimo sobre el que la aplicación está optimizada es iOS 7.0, cualquier sistema anterior puede no ejecutarse correctamente y producir anomalías.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 103: Requerimiento de Sistema No Funcional 16.**

### Identificador: RSNF - 17

Nombre:	Memoria mínima.
Descripción:	El sistema pesa un total de 8'5 Mb, debemos de tener un mínimo de 10 Mb de espacio en disco para que la aplicación funcione correctamente.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Analista.
Versión:	1.0.0
Fecha:	4/3/2015

**Tabla 104: Requerimiento de Sistema No Funcional 17.**

### 3.4 MATRIZ DE TRAZABILIDAD.

Tras analizar todos los requisitos, hemos de verificar que entre ellos tienen una relación, es decir todo requisito de usuario debe de complementarse con un requisito de sistema para que la aplicación tenga coherencia.

Para facilitar la tarea, se utilizan matrices de trazabilidad, son tablas con un eje vertical donde se posicionan los requisitos de usuario y un eje horizontal donde se posicionan los requisitos de sistema.

Es muy importante recordar que todo requisito de Usuario necesita estar ligado al menos a un requisito de sistema, si nos encontramos un caso en que un requisito queda huérfano y no está ligado a ningún otro, quiere decir que dicho requisito es inconsistente y pueden ocurrir dos situaciones, primera, el proyecto está mal analizado o le faltan detalles por cerrar o segunda, se están introduciendo mejoras o versiones posteriores en el análisis inicial.

En cualquiera de los dos casos anteriores, debemos percibir el error y subsanarlo.

	RSF - 01	RSF - 02	RSF - 03	RSF - 04	RSF - 05	RSF - 06	RSF - 07	RSF - 08	RSF - 09	RSF - 10	RSF - 11	RSF - 12	RSF - 13	RSF - 14	RSF - 15	RSF - 16	RSF - 17	RSF - 18	RSF - 19	RSF - 20
RUC-01	X	X	X																	
RUC - 02				X																
RUC - 03					X															
RUC - 04						X														
RUC - 05						X		X												X
RUC - 06							X													
RUC - 07									X											
RUC - 08									X											
RUC - 09									X											
RUC - 10									X	X										
RUC - 11									X											
RUC - 12											X	X								

RUC – 13										X		X								
RUC – 14													X							X
RUC – 15													X							
RUC – 16														X						
RUC – 17															X					X
RUC – 18																X				
RUC – 19																	X	X		X
RUC – 20																		X		
RUC – 21																	X			
RUC – 22																			X	
RUC – 23					X												X	X		
RUC – 24					X															

Tabla 105: Matriz RUC – RSF.



	RSNF - 01	RSNF - 02	RSNF - 03	RSNF - 04	RSNF - 05	RSNF - 06	RSNF - 07	RSNF - 08	RSNF - 09	RSNF - 10	RSNF - 11	RSNF - 12	RSNF - 13	RSNF - 14	RSNF - 15	RSNF - 16	RSNF - 17
RUC - 01	X																
RUC - 02	X																
RUC - 03		X															
RUC - 05			X														
RUC - 06			X														
RUC - 07				X													
RUC - 08				X													
RUC - 09				X													
RUC - 10					X												
RUC - 11				X													
RUC - 12						X											
RUC - 13							X										

RUC – 14								X									
RUC – 15								X									
RUC – 16									X								
RUC – 17										X							
RUC – 18											X						
RUC – 19												X					
RUC – 20													X				
RUC – 21												X					
RUC – 22														X			
RUC – 23															X		
RUC – 24		X															

Tabla 106: Matriz RUC – RSNF.

	RSF - 1	RSF - 8	RSF - 9	RSF - 10	RSF - 17	RSF - 19	RSNF - 1	RSNF - 3	RSNF - 5	RSNF - 12	RSNF - 14	RSNF - 16	RSNF - 17
RUR - 01	X						X						
RUR - 02		X						X					
RUR - 03			X					X					
RUR - 04						X					X		
RUR - 05		X			X			X		X			
RUR - 06				X					X				
RUR - 07	X						X						
RUR - 08												X	X

**Tabla 107: Matriz RUR – RSF/RSNF .**

Tras el análisis de la matriz podemos afirmar que el análisis de los requisitos fue satisfactorio y que la aplicación es coherente y sin defectos analíticos iniciales.

Cada requisito de usuario está cubierto por al menos un requisito de sistema lo que concluye que el ¿Qué debe de hacerse? está cubierto por el ¿Cómo debe de hacerse?.

Por lo tanto se implementarán todos los requisitos en el lanzamiento de la aplicación.



## CAPÍTULO 4: DISEÑO DEL SISTEMA.

El diseño es uno de los tres pilares básicos para poder desarrollar una aplicación. En este apartado se analiza el diseño de la arquitectura, la base de datos y su estructura así como las interfaces de usuario que tendrá la aplicación.

Como se puede intuir, el modelo MVC o vista controlador estará muy presente en nuestro diseño, donde las vistas englobarán la parte de las interfaces, la arquitectura relacionará los modelos con los controladores y la base de datos será el almacenamiento masivo de información de donde se nutrirá el controlador para mostrar la información en las vistas.

Así pues procedemos con el diseño de la arquitectura.

## 4.1 DISEÑO DE LA ARQUITECTURA

La aplicación utilizará una arquitectura basada en el diseño Cliente Servidor, esta arquitectura se basa en entablar una comunicación entre un cliente y un servidor, por lo general el número de clientes suele ser mucho más elevado que el de servidores. Los clientes tienen una aplicación, en nuestro caso será Campus Life y mediante órdenes internas de la aplicación conectan con el servidor para recibir y enviar datos, en general hacer peticiones y recibir respuestas.

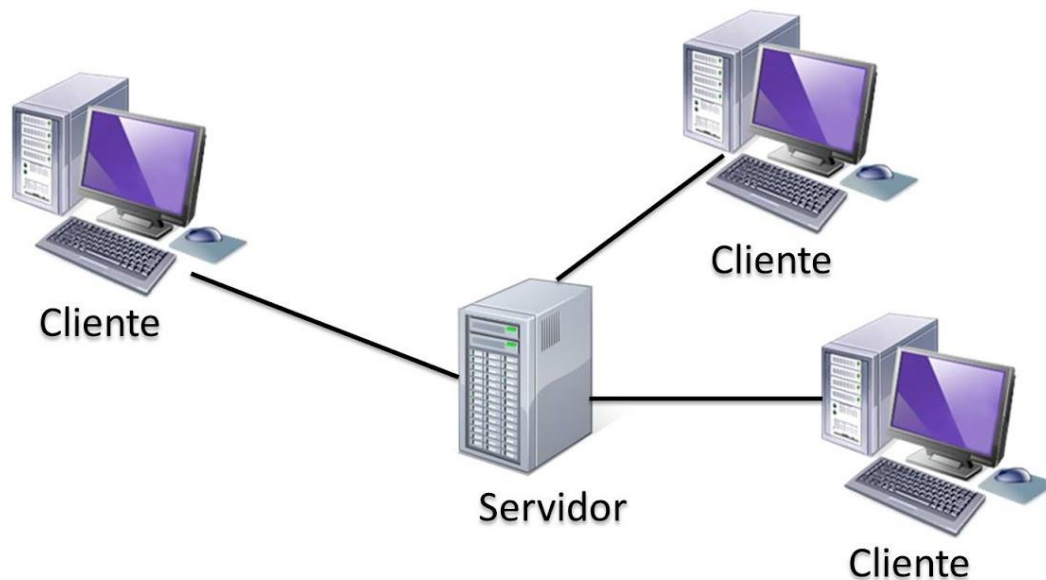
En nuestro caso, la arquitectura constará de tantos clientes como descargas de la aplicación se realicen y un servidor central situado en el campus de Getafe de la Universidad Carlos III de Madrid.

Por lo general existen dos tipos de peticiones que puede generar el cliente y que están definidos en la propia llamada, las peticiones de datos que son ligeras y por lo general no incluyen parámetros y mediante las cuales el servidor proporciona información al cliente, y las peticiones de inserción que son más pesadas y siempre requieren de parámetros de entrada con los cuales el servidor generará modificaciones en la base de datos, también puede retornar información.

### 4.1.1 PROCESO DE PETICIONES

En nuestro caso las peticiones se realizan en 3 pasos:

- **Paso 1:** La aplicación del cliente recibe una interacción y esta desencadena una llamada al servidor ya sea del tipo dato o inserción.
- **Paso 2:** Existe un intermediario del cual hablaremos posteriormente que hace de interlocutor entre el cliente y el servidor de datos, se trata de un servidor web que facilita las comunicaciones y la estabilidad del sistema y se encarga de generar la lógica de negocio.
- **Paso 3:** El servidor de datos recibe la información y proporciona los datos necesarios.



**Ilustración 21: Cliente servidor.**

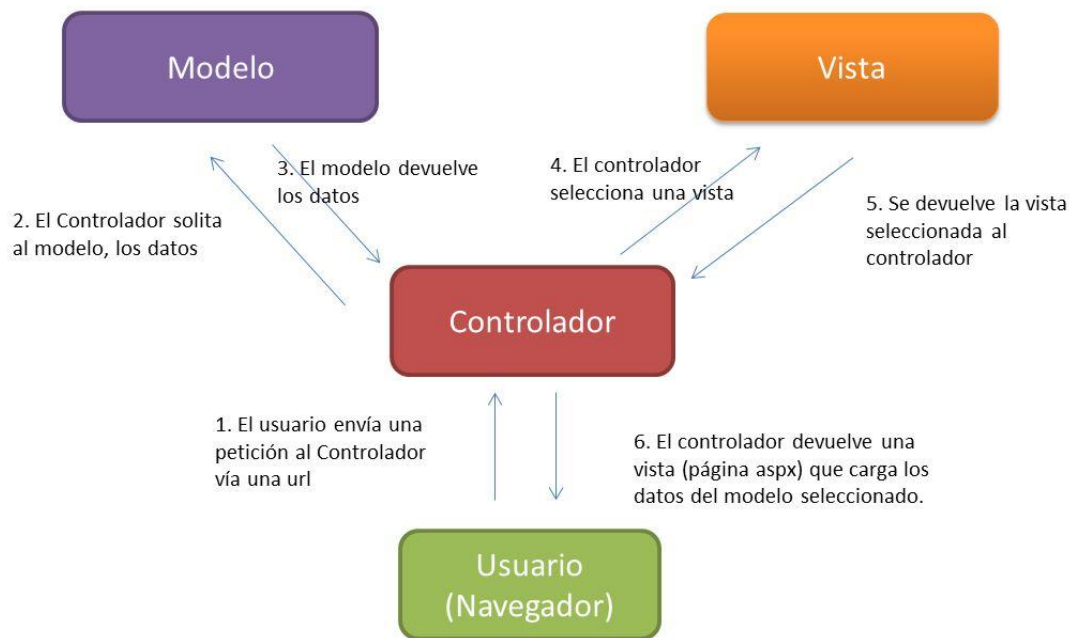
Ahora podemos observar mejor el esquema MVC al que nos referíamos anteriormente.

Como se puede comprobar el patrón Modelo Vista Controlador se adapta de manera adecuada al diseño de la arquitectura Cliente Servidor.

Para comprender mejor esta complementación entre patrón y arquitectura analizaremos las tres partes básicas del patrón.

#### 4.1.2 PATRÓN MVC

- **Modelo:** es una representación de los objetos del mundo físico, trata de obtener las propiedades y las funciones que puede realizar un objeto y las representa de manera virtual en forma de clases. Suele definirse con estructuras de datos y sus funciones suelen modificar estas estructuras.
- **Vista:** Es literalmente la representación virtual de las interfaces que el cliente va a poder manejar con la aplicación. Estas vistas contienen una serie de eventos que el usuario puede activar para comunicarse con los controladores.
- **Controlador:** se encargan de aplicar la lógica de negocio, tratan los eventos que activa el usuario desde la interfaz y aplica las funciones necesarias que modifican los datos del modelo.



**Ilustración 22: MVC.**

#### 4.1.3 INTEGRACIÓN MVC – CLIENTE SERVIDOR

A continuación se detalla el proceso paso a paso para comprender mejor el funcionamiento y la dinámica de ejecución del proceso de conexión entre cliente y servidor.

- Primero el usuario que ha descargado la aplicación o cliente de servicio, realiza una interacción con una vista de la aplicación, una de las interfaces del sistema, esto desencadena un evento que en nuestro caso se trata de una petición a servidor, hay muchos más tipos de eventos que no generan comunicación.
- Seguidamente el controlador recibe los datos de la llamada y dependiendo del tipo de petición y de los parámetros de la misma los analiza y los modifica en una de sus funciones.
- Estos datos enviados se relacionan directamente con un objeto definido en el modelo y la función puede leer, escribir o modificar dicho objeto.
- El controlador juega con los datos y los objetos instanciados de las clases pero no almacena ningún tipo de dato, esto lo hace la base de datos por lo que el controlador tiene que comunicar con la base de datos y afianzar los cambios o modificaciones.



- La base de datos devuelve, escribe o modifica los datos dependiendo de lo indicado por el controlador y el controlador vuelve a utilizar dichos datos para aplicar de nuevo la lógica pertinente.
- Por último el controlador envía una respuesta a la petición de la vista y se muestra al usuario el resultado final de los datos.

No siempre se producen todas las comunicaciones, puede que la propia vista ejecute un evento que no necesita del controlador, o que el controlador pueda modificar los datos de la vista sin necesidad de conectar con la base de datos, o que incluso la vista no reciba ninguna respuesta del controlador.

En el caso de nuestra aplicación la parte controladora se ha generado mediante un Web Service, como se comentó con anterioridad, estos servicios nos facilitan las comunicaciones utilizando los protocolos http de comunicación y estandarizando las comunicaciones, además liberan a la vista de comunicaciones más complejas y pueden utilizarse para varios servicios simultáneos, en nuestro caso la mayoría de peticiones son del tipo GET y únicamente piden información para mostrar al usuario final, por lo que la lógica de negocio no es complicada y básicamente se dedica a comunicar la vista con la base de datos.

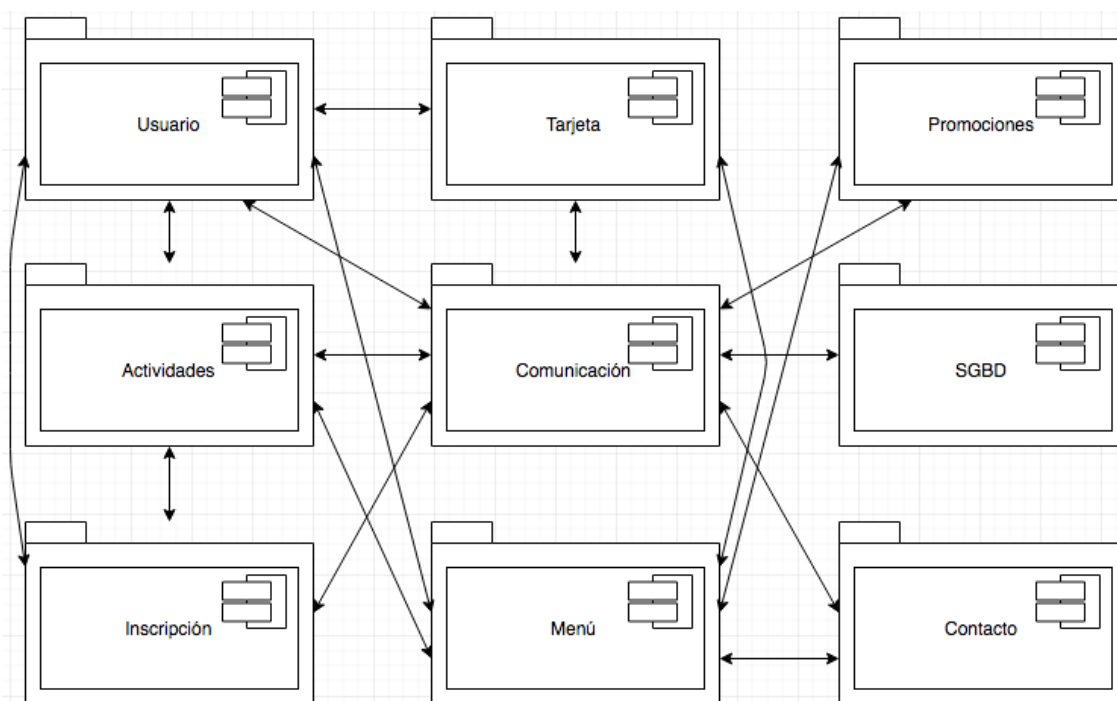
#### 4.1.4 LISTA DE PETICIONES

Todas las llamadas pueden realizarse desde el entorno de producción o el entorno de pruebas y su construcción es similar salvo la primera ruta de la URL que varía dependiendo del entorno en que deseemos ejecutar la aplicación. Esto evidentemente solo afectará a los desarrolladores de la aplicación y los testers que la prueben antes de salir al mercado, donde todas las llamadas de los clientes se realizarán al entorno de explotación.

- **Llamada de Actividades:** pide una lista de actividades indicándole al controlador cual es el campus del que debe obtener las actividades. Retorna una lista de actividades del campus seleccionado.
- **Llamada de Actividad:** pide una actividad indicando el código de la actividad al controlador. Retorna un objeto del tipo Actividad.
- **Llamada de Actividades inscritas:** pide una lista de actividades indicándole al controlador el usuario que la solicita. Retorna una lista de actividades.

- **Llamada de Inscripción:** pide al controlador una respuesta sobre la posible inscripción en una cierta actividad, le indica el código de la actividad. El controlador retorna la posibilidad de inscribirse o de apuntarse a una lista de espera.
- **Llamada de Inscripción lista de espera:** pide al controlador una respuesta sobre la posible inscripción en la lista de espera de una cierta actividad, le indica el código de la actividad. El controlador retorna la posibilidad de apuntarse a la lista de espera.
- **Llamada de Datos Bancarios existentes:** se pregunta al controlador si hay datos bancarios almacenados en base de datos del cliente.
- **Llamada de Datos Bancarios:** le manda al controlador los datos bancarios necesarios para realizar la operación de inscripción posteriormente.
- **Llamada de Tarjeta TE:** el controlador manda una lista de las posibles tarjetas que el usuario puede adquirir.
- **Llamada de Comprar tarjeta TE:** se indica al controlador el tipo de tarjeta que se desea comprar mandándole el id de la misma.
- **Llamada de Datos del usuario:** se pide al controlador los datos del usuario que se ha logueado para mostrarlos en el menú.
- **Llamada de Login:** el usuario manda sus credenciales al controlador y este verifica su identidad en la base de datos. Retorna un token de usuario en caso de éxito y un error en caso contrario.
- **Llamada de Refrescar Token:** cuando ha transcurrido el tiempo de vida del token y el usuario sigue conectado, se utiliza un token de refresco para obtener un nuevo token sin necesidad de volver a introducir las credenciales. Se manda al controlador un token de refresco y se recibe un token nuevo.
- **Llamada de Promociones:** se reciben todas las promociones en una lista.

#### 4.1.5 COMPONENTES DE LA ARQUITECTURA



**Ilustración 23: Arquitectura.**

Cada componente del anterior diagrama se encarga de realizar un tipo de peticiones y establecer comunicaciones para un correcto transito de los datos, vamos a definir una tabla que describa qué función realiza cada componente y la relación que tiene con los requisitos.

**Identificador: COM - XX**

Nombre:	
Propósito:	
Función:	
Relación:	

**Tabla 108: Descripción Tabla de Componentes.**

- **Nombre:** nombre identificativo del componente.
- **Identificador:** identificador unívoco del componente. Su nomenclatura ha de seguir el formato: COM\_XX, donde XX será sustituido por un número comenzando en 1.
- **Finalidad:** propósito que pretende alcanzar el componente en la aplicación.

- **Función:** listado de funcionalidades que llevará a cabo el componente en el sistema.
- **Relación:** indica qué requisitos de usuario cubre este componente. Es importante comprobar que todos los requisitos queden cubiertos por los componentes.

### Identificador: COM - 01

Nombre:	Componente Actividades.
Finalidad:	Mostrar las actividades ofertadas por espacio estudiantes y revisar todos su parámetros para el correcto funcionamiento.
Función:	<ul style="list-style-type: none"> <li>• Mostrar el nombre la actividad.</li> <li>• Mostrar la descripción de la actividad.</li> <li>• Mostrar una imagen de la actividad.</li> <li>• Mostrar botón de reproducir sobre la actividad.</li> <li>• Mostrar precios de la actividad.</li> <li>• Mostrar la opción de inscripción en la actividad.</li> <li>• Mostrar la opción de horario de la actividad.</li> <li>• Marcar o desmarcar como favorita una actividad.</li> <li>• Clasificar una actividad según su sección.</li> <li>• Compartir una actividad.</li> <li>• Revisar actividades inscritas, para visualizar o no.</li> <li>• Obtener el horario de una actividad.</li> <li>• Clasificar las actividades por campus.</li> <li>• Obtener las actividades inscritas.</li> <li>• Obtener las actividades favoritas.</li> <li>• Borrar actividades favoritas.</li> </ul>
Relación:	RSF – 07, 08, 09, 10.

Tabla 109: Componente de Actividad.

### Identificador: COM - 02

Nombre:	Componente Promociones.
Finalidad:	Mostrar las promociones ofertadas por espacio estudiantes y permitir al usuario disfrutar de los descuentos que ofrecen las tarjetas TE.
Función:	<ul style="list-style-type: none"> <li>• Mostrar el nombre la promoción.</li> <li>• Mostrar la descripción de la promoción.</li> <li>• Mostrar una imagen de la promoción.</li> <li>• Compartir una promoción.</li> <li>• Clasificar las ofertas según personal.</li> </ul>

	<ul style="list-style-type: none"> <li>• Abrir dirección de enlace externo.</li> <li>• Compartir.</li> </ul>
Relación:	RSF - 17, 18.

**Tabla 110: Componente de Promociones.**

**Identificador: COM - 03**

Nombre:	Componente Tarjetas.
Finalidad:	Mostrar las tarjetas ofertadas por espacio estudiantes y permitir la compra de la mismas por parte del usuario.
Función:	<ul style="list-style-type: none"> <li>• Mostrar el nombre las tarjetas.</li> <li>• Mostrar la descripción de las tarjetas.</li> <li>• Clasificar las tarjetas según campus.</li> <li>• Revisar datos del usuario para la compra.</li> <li>• Añadir número de cuenta.</li> <li>• Modificar número de cuenta.</li> <li>• Realizar compra de tarjeta.</li> <li>• Compartir.</li> </ul>
Relación:	RSF – 11, 12.

**Tabla 111: Componente de Tarjetas.**

**Identificador: COM - 04**

Nombre:	Componente Inscripción.
Finalidad:	Permitir al usuario realizar la inscripción de una actividad.
Función:	<ul style="list-style-type: none"> <li>• Mostrar el nombre la actividad.</li> <li>• Mostrar el precio final de la actividad.</li> <li>• Revisar datos de usuario para la inscripción.</li> <li>• Añadir número de cuenta.</li> <li>• Modificar número de cuenta.</li> <li>• Realizar inscripción.</li> <li>• Compartir.</li> </ul>
Relación:	RSF – 15, 16.

**Tabla 112: Componente de Inscripción.**

**Identificador: COM - 05**

Nombre:	Componente Usuario.
Finalidad:	Mostrar los datos del usuario, y utilizar sus credenciales de login para

	inicializar la aplicación con un token que permita hacer peticiones.
Función:	<ul style="list-style-type: none"> <li>• Mostrar el nombre del usuario.</li> <li>• Comprobar credenciales.</li> <li>• Conseguir Token.</li> <li>• Crear estructuras internas de almacenamiento.</li> <li>• Borrar estructuras internas de almacenamiento.</li> </ul>
Relación:	RSF – 01, 02, 03, 04.

**Tabla 113: Componente de Usuario.**

**Identificador: COM - 06**

Nombre:	Componente Contacto.
Finalidad:	Mostrar las diferentes maneras de contactar de ESES.
Función:	<ul style="list-style-type: none"> <li>• Contactar mediante RRSS.</li> <li>• Contactar mediante correo electrónico.</li> <li>• Llamar a las oficinas de ESES.</li> <li>• Mostrar localización de las oficinas de ESES.</li> <li>• Abrir un mapa.</li> <li>• Compartir.</li> </ul>
Relación:	RU – 18,19,20,21.

**Tabla 114: Componente de Contacto.**

**Identificador: COM - 07**

Nombre:	Componente Menú.
Finalidad:	Mostrar los servicios que ofrece la aplicación.
Función:	<ul style="list-style-type: none"> <li>• Mostrar nombre de usuario conectado.</li> <li>• Salir de la sesión.</li> <li>• Acceso a compra de TE.</li> <li>• Acceso a vista de Actividades.</li> <li>• Acceso a vista de Favoritas.</li> <li>• Acceso a vista de Inscritas.</li> <li>• Acceso a vista de Contacto.</li> <li>• Acceso a vista de Promociones.</li> <li>• Acceso a vista de Información.</li> </ul>
Relación:	RU – 22,23,24.

**Tabla 115: Componente de Menú.**

#### 4.1.6 MATRIZ DE TRAZABILIDAD

Tras analizar todos los componentes, hemos de verificar que todos los requisitos tienen una relación directa con ellos, es decir todo requisito de usuario debe de complementarse con un componente de la arquitectura lo cual asegura que se cumplirán las funcionalidades requeridas.

	RSF - 01	RSF - 02	RSF - 03	RSF - 04	RSF - 05	RSF - 06	RSF - 07	RSF - 08	RSF - 09	RSF - 10	RSF - 11	RSF - 12	RSF - 13	RSF - 14	RSF - 15	RSF - 16	RSF - 17	RSF - 18	RSF - 19	RSF - 20
COM - 01						X	X	X	X				X	X	X	X				X
COM - 02																	X	X		
COM - 03											X	X								
COM - 04										X										
COM - 05	X	X	X	X																
COM - 06																			X	
COM - 07					X															

Tabla 116: Matriz COM - RSF

## 4.2 DISEÑO DE LA BASE DE DATOS.

No toda la información que el usuario puede ver en la interfaz cliente es dinámica, esto quiere decir que siempre hay datos estáticos que puede almacenar el propio cliente sin necesidad de ser consultados a un servidor externo, pero la gran mayoría de información que se visualiza en la actualidad en cualquier aplicación es dinámica, es decir, cambia con el tiempo y eso es un pequeño hándicap puesto que no se puede almacenar en el cliente.

Para ello la conectividad es una pieza fundamental, pues a la hora de actualizar o necesitar cualquier tipo de dato o información dinámica debemos de entablar una comunicación o conexión con un servidor que la administre.

Como punto positivo, las aplicaciones son muy ligeras, como punto negativo, dependemos exclusivamente de una conexión a la red que nos suministre los datos. A día de hoy esto no es un gran problema pues la globalización y la interconectividad entre dispositivos ha avanzado de manera desmesurada.

Para almacenar y centralizar todos estos datos se utilizan las Bases de Datos que hacen un almacenamiento masivo de información y que blindan de seguridad los datos del usuario puesto que únicamente el servidor puede acceder a ellos, por otro lado cualquier error en la base de datos afectaría a todos los usuarios que usan el cliente.

En Campus Life se ha determinado utilizar una base de datos relacional popular como es MySQL, un gestor de base de datos de código abierto.

Campus Life se distribuye a través de los markets de Google y Apple y cada vez que uno de los clientes necesitan información, la solicitan al servidor, que en este caso será un servidor de aplicaciones que mediante Web Services se comunica con el cliente y mediante una conexión directa con la base de datos a través del lenguaje SQL puede leer, escribir o modificar, lo que en la actualidad se denomina CRUD.

En la siguiente imagen se puede apreciar el modelo relacional que cumple nuestra base de datos, a continuación se detallarán cada una de las entidades y sus relaciones.



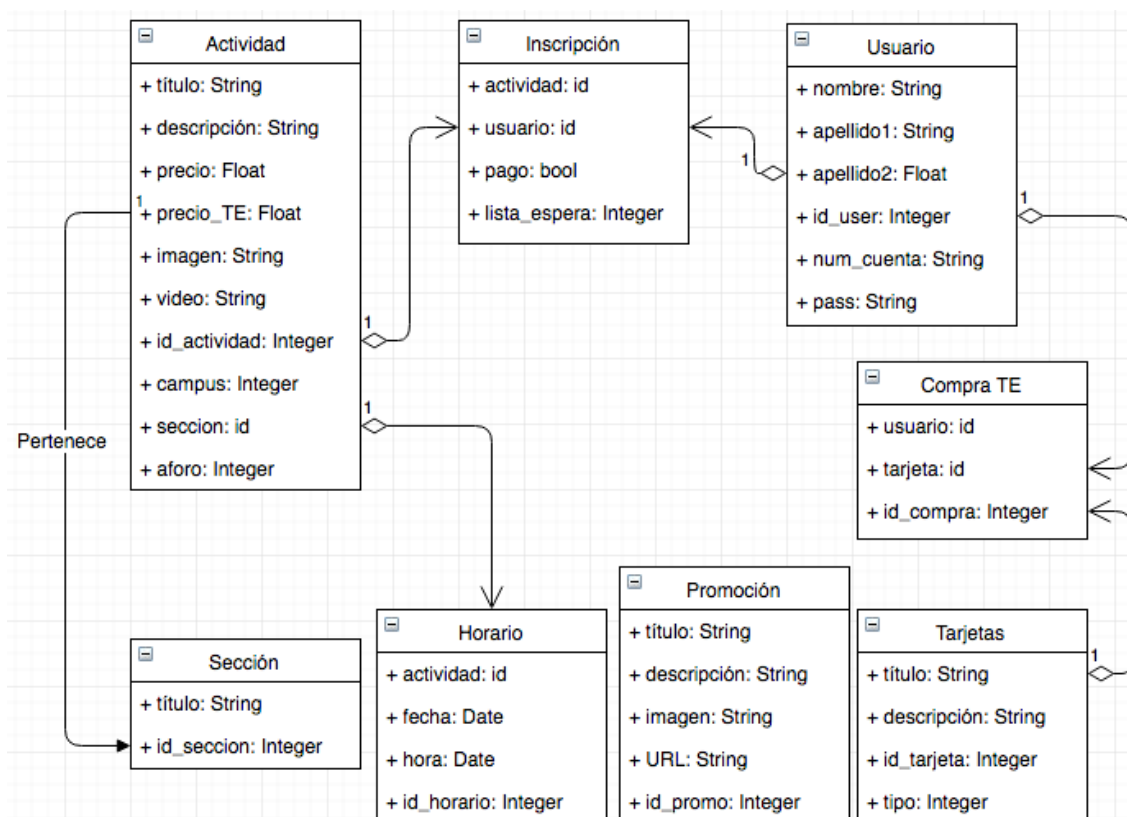


Ilustración 24: Modelo Relacional de la BBDD.

#### 4.2.1 ENTIDAD ACTIVIDAD.

Es la entidad principal de la aplicación y la que ocupa mayor tamaño, consta de diversos atributos que definen una actividad.

- **id\_actividad:** es una clave única que identifica de manera unívoca cada entrada de la tabla.
- **sección:** es una clave única que relaciona nuestra actividad con un elemento de la tabla Sección.
- **título:** es el nombre de la actividad.
- **descripción:** se trata de la descripción de la actividad, en qué consiste y como se desarrolla la misma.
- **precio:** es el precio que cuesta la inscripción de la actividad sin TE.
- **precio\_te:** es el precio que cuesta la inscripción de la actividad con TE.
- **imagen:** es una URL de la dirección donde se aloja la imagen.
- **video:** es una URL de la dirección donde se aloja el video.
- **campus:** es un número entero entre 0 y 3 que define el campus al que pertenece la actividad.
- **aforo:** es un número entero que define el máximo número de personas que se pueden apuntar a dicha actividad.

La tabla Actividad tiene un uso exclusivo de lectura, puesto que el cliente no puede modificar los datos de la tabla ni crear nuevas entradas, los administradores de la base de datos pueden realizar escrituras y modificaciones pero siempre en casos puntuales para corregir errores o añadir nuevas actividades en las fechas adecuadas.

#### 4.2.2 ENTIDAD USUARIO.

Es la segunda entidad principal y define y almacena datos de los usuarios que van a identificarse en la aplicación.

- **id\_user:** es una clave única que identifica de manera unívoca cada entrada de la tabla.
- **pass:** se trata de la contraseña del usuario cifrada.
- **nombre:** se trata del nombre del usuario.
- **apellido1:** primer apellido del usuario.
- **apellido2:** segundo apellido del usuario.
- **num\_cuenta:** número de la cuenta bancaria del usuario.

La tabla Usuario tiene acceso exclusivo de lectura en todos sus campos excepto en num\_cuenta que podrá ser modificado o añadido por el usuario en caso de no estar registrado.

#### 4.2.3 ENTIDAD INSCRIPCIÓN.

Es una tabla muy importante puesto que almacena todas las inscripciones que realizan los usuarios de las actividades. Además el número de entradas de una actividad en concreto se utilizarán para contrastar el aforo de una actividad, es un dato implícito almacenado.

- **actividad:** es una clave única que relaciona la tabla Actividades con Inscripción, forma junto a usuario parte de la clave primaria.
- **usuario:** es una clave única que relaciona la tabla Usuario con Inscripción, forma junto a actividad parte de la clave primaria.
- **pago:** se trata de un booleano que indica si el pago está realizado.
- **lista\_espera:** es una inscripción en espera, si tiene un 0 significa que la actividad está inscrita, si la actividad tiene un número, indica la posición en la lista de dicha actividad.

La combinación de actividad y usuario forma la clave primaria de la tabla y no puede haber una combinación de ambos campos que coincida.

La tabla Inscripción tiene acceso de escritura y lectura, nunca de modificación, cuando un usuario se inscribe en una actividad se inicializa una nueva fila con los datos de la actividad inscrita. También el cliente hace lecturas de la tabla para la sección de Mis Actividades donde se debe de mostrar las actividades inscritas de un usuario, y en la sección de Actividades para no mostrar a un usuario las actividades en las que ya se encuentra inscrito.

#### 4.2.4 ENTIDAD SECCIÓN.

La tabla de sección sirve para dar la propiedad nombre a la sección, podría haberse utilizado el de cada sección para almacenarlo en la columna sección de la tabla Actividades, pero es mejor trabajar con enteros a la hora de identificar elementos.

- **título:** es el nombre de la sección.
- **Id\_sección:** es una clave única que identifica de manera unívoca cada entrada de la tabla.

La tabla Sección tiene un uso exclusivo de lectura, puesto que el cliente no puede modificar los datos de la tabla ni crear nuevas entradas, los administradores de la base de datos pueden realizar escrituras y modificaciones pero siempre en casos puntuales para corregir errores o añadir nuevas secciones en las fechas adecuadas.

#### 4.2.5 ENTIDAD HORARIO.

La tabla de sección sirve para dar la propiedad fecha y hora a la actividad, podría haberse pensado utilizar un par de columnas en la tabla Actividad para almacenar la fecha y la hora, pero como una actividad puede tener varias fechas y horas se repetirían las filas con el mismo identificador de actividad.

- **actividad:** es una clave única que relaciona la tabla Actividades con Horario.
- **Id\_horario:** es una clave única que identifica de manera unívoca cada entrada de la tabla.
- **fecha:** indica una de las fechas en las que la actividad se impartirá.
- **hora:** indica una hora en la que se impartirá la actividad.

La tabla Horario tiene un uso exclusivo de lectura, puesto que el cliente no puede modificar los datos de la tabla ni crear nuevas entradas, los administradores de la base de datos pueden realizar escrituras y modificaciones

pero siempre en casos puntuales para corregir errores o añadir nuevas actividades en las fechas adecuadas.

#### 4.2.6 ENTIDAD PROMOCIÓN.

Es una tabla sin relaciones, almacena datos de las promociones que oferta ESES.

- **id\_promocion:** es una clave única que identifica de manera unívoca cada entrada de la tabla.
- **título:** es el nombre de la promoción.
- **descripción:** se trata de la descripción de la promoción, en qué consiste y como se desarrolla la misma.
- **imagen:** es una URL de la dirección donde se aloja la imagen.
- **URL:** es un número entero entre 0 y 3 que define el campus al que pertenece la actividad.

La tabla Promoción tiene un uso exclusivo de lectura, puesto que el cliente no puede modificar los datos de la tabla ni crear nuevas entradas, los administradores de la base de datos pueden realizar escrituras y modificaciones pero siempre en casos puntuales para corregir errores o añadir nuevas promociones en las fechas adecuadas.

#### 4.2.7 ENTIDAD COMPRA TE.

Es una lista de tarjetas que han sido compradas por los usuarios.

- **id\_compra:** es una clave única que identifica de manera unívoca cada entrada de la tabla.
- **usuario:** es una clave única que relaciona la tabla Usuario con Compra TE.
- **tarjeta:** es una clave única que relaciona la tabla Tarjetas con Compra TE.

La tabla Compra TE tiene acceso de escritura y lectura, nunca de modificación, cuando un usuario se compra una tarjeta se inicializa una nueva fila con los datos de la tarjeta comprada. También el cliente hace lecturas de la tabla para la sección de Tarjeta TE para no mostrar a un usuario las tarjetas que ya ha comprado.

#### 4.2.8 ENTIDAD TARJETAS.

Es una tabla con los tipos de tarjeta que ESES oferta para su compra.

- **id\_tarjeta:** es una clave única que identifica de manera unívoca cada entrada de la tabla.
- **tipo:** indica el tipo de tarjeta.
- **título:** es el nombre de la tarjeta.
- **descripción:** se trata de la descripción de la tarjeta y de sus beneficios.

La tabla Tarjeta tiene un uso exclusivo de lectura, puesto que el cliente no puede modificar los datos de la tabla ni crear nuevas entradas, los administradores de la base de datos pueden realizar escrituras y modificaciones pero siempre en casos puntuales para corregir errores o añadir nuevas tarjetas en las fechas adecuadas.

### 4.3 DISEÑO DE LA INTERFAZ.

El diseño de las interfaces de usuario puede que sea uno de los más importantes del desarrollo de un proyecto, los gurús de la tecnología buscan implantar las últimas novedades y sistemas para que sus aplicaciones sean punteras en el mercado, pero el análisis de mercado y de los gustos del cliente potencial son la pieza clave del éxito de muchos productos. Apple sabe mucho de esto y lo demuestra día a día, tiene una tecnología impecable, utiliza la mitad de potencia que sus competidores llegando prácticamente a los mismos resultados, pero lo que más destaca de sus productos es su diseño innovador y su estética limpia y brillante que convierte a sus productos en objetos exclusivos, aumentando sus ventas con artículos a precios por encima de la media.

Campus Life para iOS ha tenido que adaptarse a los estándares de Apple y además ha sabido introducir ese punto de limpieza y brillo con la que la compañía Americana ha revolucionado el mercado.

Campus Life se basa en un diseño de interfaz minimalista, intentado crear un espacio limpio y claro donde los objetos no se amontonen y estén claras las funcionalidades sin cargar la vista del usuario final. Su juego de colores es muy nítido y juega con los blancos para dar aspecto de limpieza, el color anaranjado es el más utilizado debido a que se trata del color corporativo para aplicaciones de última generación de la Universidad Carlos III de Madrid. El manual de estilos aplicados a la app es el siguiente:

#### **Tipografías:**

- El nombre de la aplicación que aparecerá al principio en la cabecera irá en Rotis Semi Serif.
- El resto de la tipografía será la fuente que utiliza iPhone como estándar. Helvética.
- Los textos deben ir en minúsculas siempre. (respetando la regla de que la letra de la primera palabra va en mayúsculas).
- El contenido irá en #3a3a3a, y el tono dominante de la app cuando sea necesario resaltar algún contenido (Avisos, Alertas..) será #ff5a00, irá en blanco cuando vaya sobre fondo oscuro.
- El cuerpo del contenido puede variar tamaños (11, 12, 14).
- El nombre de la app puede ir a 16pt en la portada.

#### Fondos:

- Blanco traslúcido.
- Pasamos al estándar de iOS y hacemos la app más limpia.

#### Cabeceras:

- Irán en tono #ff5a00.
- El icono que aparecerá es el icono de la app, desaparece el acrónimo de la UC3M.

#### Botones:

- Utilizar los botones del sistema operativo siempre que existan.
- Borde en el tono dominante de la app. Naranjas, blanco o gris.
- Cambio a tono dominante o gris oscuro en “over”.
- Crear estrella de Favoritas y letras de pestaña del campus.

A continuación se definen los aspectos técnicos del diseño.

#### 4.3.1 DISEÑO DE LAS VISTAS.

La vista de **Login** es la primera que el usuario ve nada más abrir la aplicación, es una vista sencilla y con 4 partes:

- El icono y nombre de la aplicación, será la única vez que aparece el nombre Campus Life en la aplicación. El logotipo se utilizará posteriormente como botón del menú principal.
- El usuario y el cuadro de texto para introducirlo, como en todos los productos UC3M se limitan los caracteres y se parsean caracteres especiales para que la comunicación entre cliente y servidor no produzca errores.
- La contraseña y el cuadro de texto para introducirla, de nuevo se limitan los caracteres y se parsean, además como en todos los campos de contraseña se visualiza la escritura con Asteriscos o puntos que nos permiten saber el número de caracteres pero no cuales son, está es la primera medida de seguridad de la aplicación.
- Label de alerta, botón de inicio y logotipo corporativo de la universidad, al igual que el nombre de la aplicación el logotipo no volverá a visualizarse en la aplicación, salvo en la pantalla de información. El label

de alerta es transparente y aparece con letras rojas cuando algo ha salido mal, si las credenciales son correctas muestra un mensaje de bienvenida pero en la mayoría de los casos es inapreciable debido a la velocidad de carga de la siguiente vista.

- Como en muchas aplicaciones, cuando los dispositivos carecen de una pantalla amplia y grande como era el caso de los iPhone 3, 4 y 4S, el teclado emergente ocupa gran parte de la pantalla y tapa los campos de texto, nuestra vista de login no tiene scroll y se ha ideado una función que desplaza los campos de texto para poder escribir correctamente cuando el teclado emerge, habría sido más fácil crear una vista scrollable pero la interacción de esta manera es más fluida y no molesta al usuario teniendo que hacer scroll, automáticamente nuestro login lo hace por ti y en cualquier tipo de iPhone.



Ilustración 25: Login.



Tras verificar las credenciales, la siguiente vista en cargar es siempre la de **Actividades**, quizás sea la vista que más información proporciona al usuario y por consecuente la que más satura la vista del usuario, la podemos dividir en 3 secciones la sección de título y menús, la sección de vista carrusel y la sección de campus de pestañas, como se puede apreciar es una vista muy compleja con gran cantidad de interacciones.

Sección Superior:

- Menú: podemos observar como hay un pequeño botón en la esquina superior izquierda con el icono de la aplicación, si pulsamos este botón se abrirá el menú lateral.
- Título de vista: En todas las vistas de la aplicación hay un título con el nombre de la vista en que nos encontramos, es muy útil puesto que en algunas vistas como Favoritas o Mis Actividades son realmente parecidas y nos ayudan a identificar donde nos encontramos navegando.

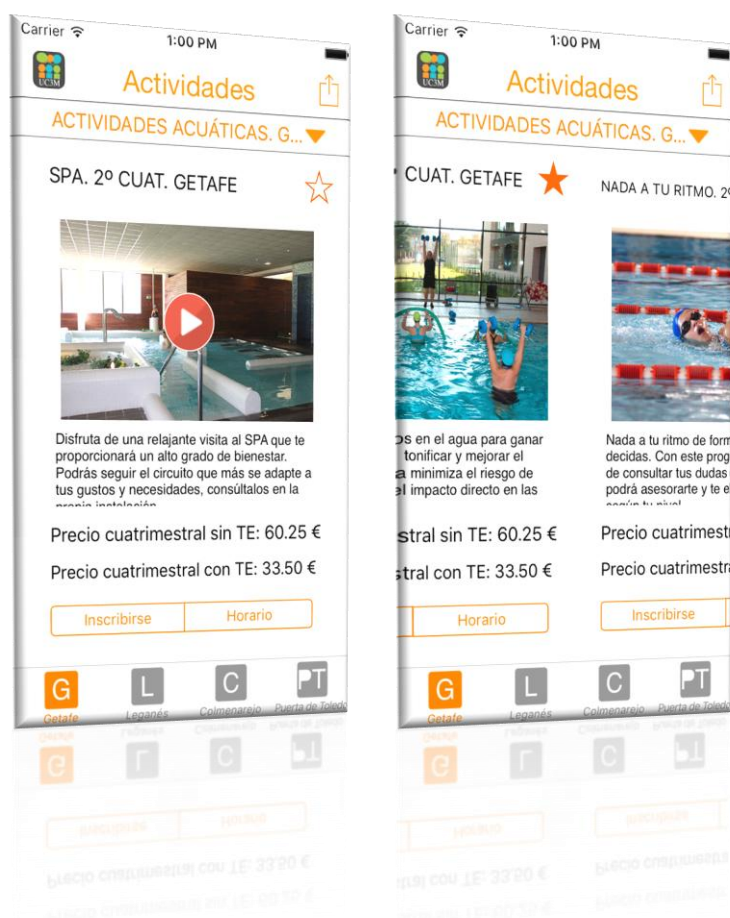


Ilustración 26: Actividades.

- Botón de compartir, en toda la aplicación encontraremos este botón, las redes sociales están a la orden del día y el acceso directo para publicar es idóneo.
- Desplegable de secciones, las actividades se agrupan en secciones y mediante este desplegable podremos ir al tipo de actividad deseada sin tener que recorrer todas las anteriores.

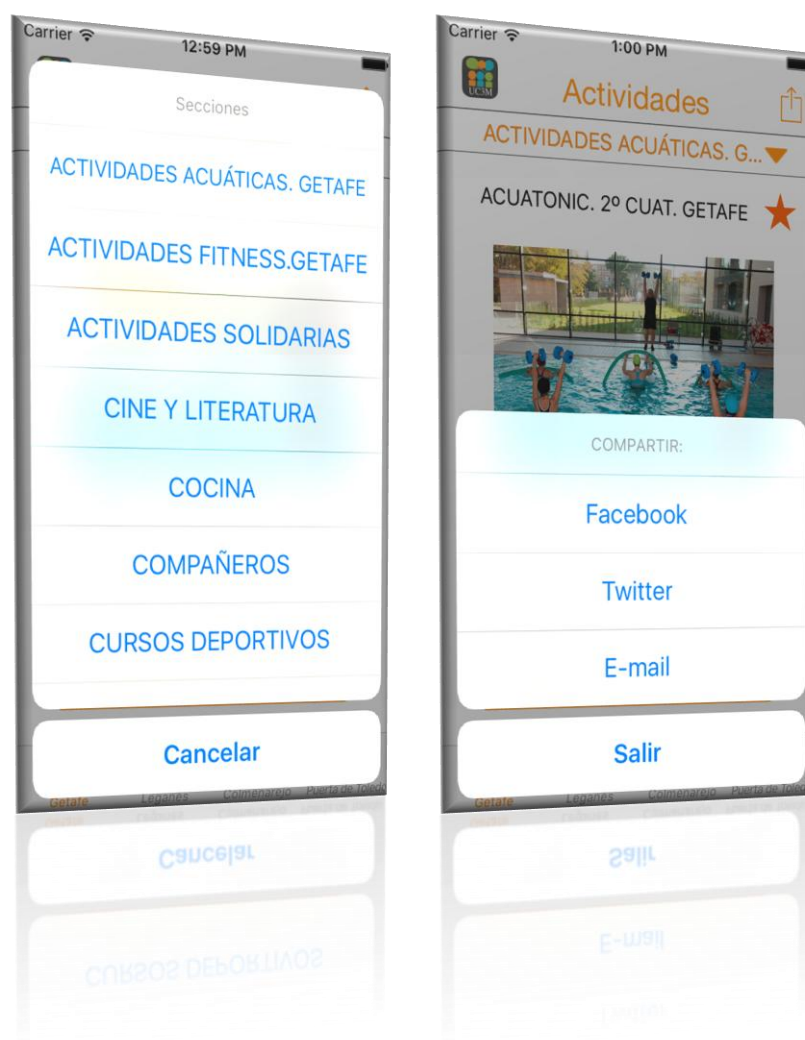
#### Sección Central (Vista Carrusel):

Se trata de una sub vista que al igual que las filas de una tabla solo crea 3 objetos sub vista que van refrescándose, de esta manera no tiene que instanciar una vista cada vez que el carrusel se mueve, únicamente refresca los datos.

- Título de la Actividad: Es lo primero que se muestra y tiene un autoajuste de tamaño, de esta manera siempre se leerá el título completo aunque en ocasiones se pierda tamaño de letra.
- Estrella de favorito: está a la derecha del título y puede marcarse o desmarcarse, cuando un usuario se loguea se crea un vector para actividades favoritas que el usuario rellena o vacía marcando y desmarcando el botón estrella.
- Imagen de la actividad: está debajo del título, muestra una imagen de la actividad y ocupa la mayor parte del espacio.
- Botón de reproducir: ocasionalmente se encuentra superpuesto sobre la imagen de la actividad, dependiendo de si la actividad tiene video o no.
- Descripción de la imagen: se trata de un cuadro de texto con scroll, debido a que en dispositivos con pantalla pequeña no cabían todos los elementos adecuadamente.
- Precios: muestra los precios con TE y sin TE.
- Botones inscribirse y horario: ambos interrumpen la vista, inscribirse realiza una transición y horario abre un pop-up.

Sección Inferior:

Pestañas(Tabs): Muestran una pestaña por cada campus que tiene la UC3M, cada pestaña carga sus Actividades e inicializan las vistas oportunas, es decir, si abrimos únicamente Getafe se instancian 3 sub vistas, si abrimos Leganés otras 3 y si volvemos a la pestaña Getafe, nos cargará las instancias ya creadas almacenadas en memoria. Carga solo la primera vez, luego cachea la información.



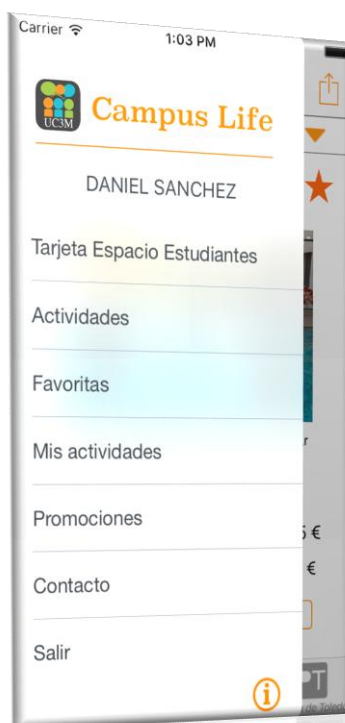
**Ilustración 27: Ventanas emergentes.**

Sobre los eventos, podemos observar como aparecen ventanas emergentes cuando compartimos o buscamos por sección, incluso si miramos los horarios, se muestra una lista con diferentes opciones. Además destacar la fluidez del carrusel y su sensibilidad que están muy logradas.

La siguiente vista es una de las más utilizadas y se puede acceder a ella desde cualquiera de las demás vistas, se trata de nuestro **Menú** principal, la raíz de la navegación y punto de partida para todas las funciones de nuestra aplicación.

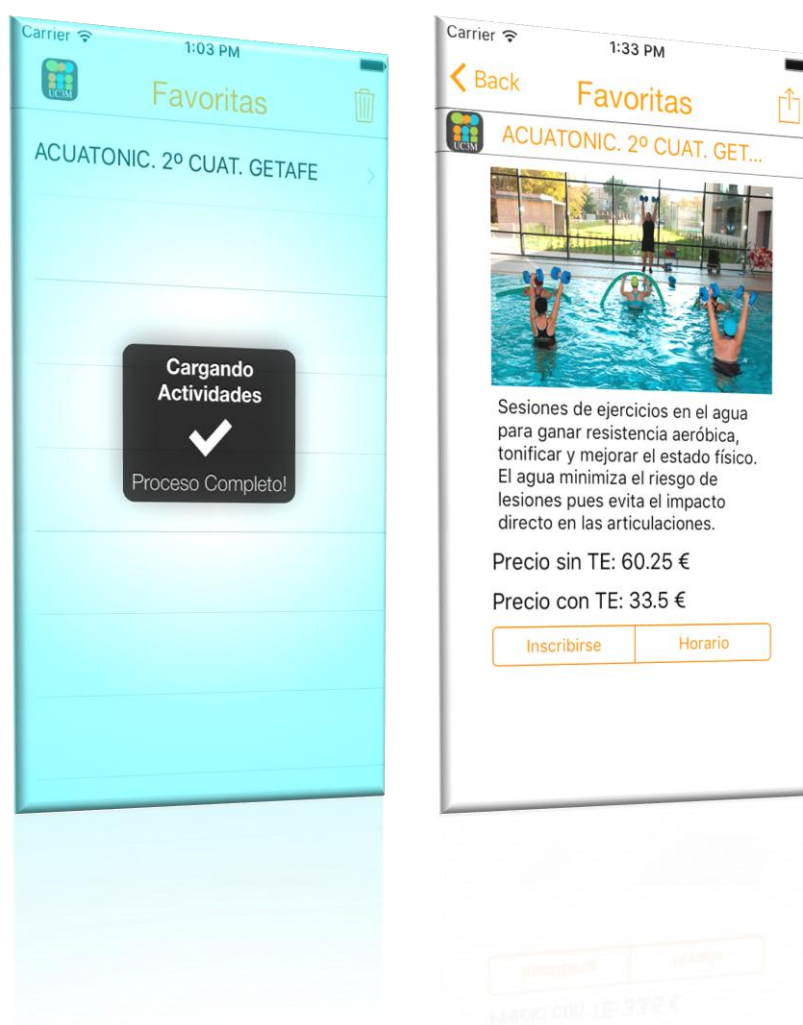
Como podemos observar en la imagen, el menú consta de una lista de funcionalidades, y datos del usuario conectado, esta información es útil para saber en todo momento con que sesión hemos abierto la aplicación. Las opciones son las siguientes:

- Tarjeta Espacio Estudiantes: permite comprar la tarjeta de ESES.
- Actividades: permite ver las actividades que se ofertan en cada campus.
- Favoritas: permite ver una lista de las actividades favoritas.
- Mis Actividades: permite ver una lista de las actividades en que nos encontramos inscritos.
- Promociones: permite ver las promociones ofertadas por ESES.
- Contacto: permite llamar y geo-localizar las oficinas de ESES en cada campus.
- Salir: redirige a la pantalla de Login y borra el token de comunicación.
- Información: muestra créditos de desarrollo y diseño junto al número de versión de la aplicación.



Cabe destacar que la vista del Menú o Drawer se denomina de esta manera porque es deslizante y se esconde y aparece según las necesidades del usuario, son muy útiles cuando hay falta de espacio en la pantalla.

Pasamos ahora a la vista de **Favoritas**, una vista que se basa en una lista de celdas que incluyen el nombre de las actividades marcadas como favoritas en la vista de Actividades, estas celdas son pulsables y transitarán a otra vista que nos muestra en detalle los datos de la actividad, como podemos observar en la 4ª imagen de Favoritas, la vista es muy parecida a la de Actividades, evidentemente se aprecia la falta de algunos elementos chocantes como la estrella de Favoritas o las pestañas de campus. Sin embargo se mantienen el resto de funcionalidades, de esta manera podemos ojear de un vistazo todas las actividades favoritas y si tenemos interés en obtener más información de una actividad en concreto, pulsar la celda y transitar hacia la vista del detalle.



**Ilustración 28: Favoritas.**

Volviendo a la primera imagen, podemos observar un pequeño detalle que tienen todas las vistas, se denomina HUD y se trata de sistemas de aviso al usuario mientras carga una vista todos sus elementos, en ocasiones son tan rápidas las cargas que no se aprecia bien el HUD pero cuando la conexión es mala podemos observar como el HUD nos avisa del proceso de carga y nos

informa de que todo está correctamente y la aplicación no ha quedado bloqueada.

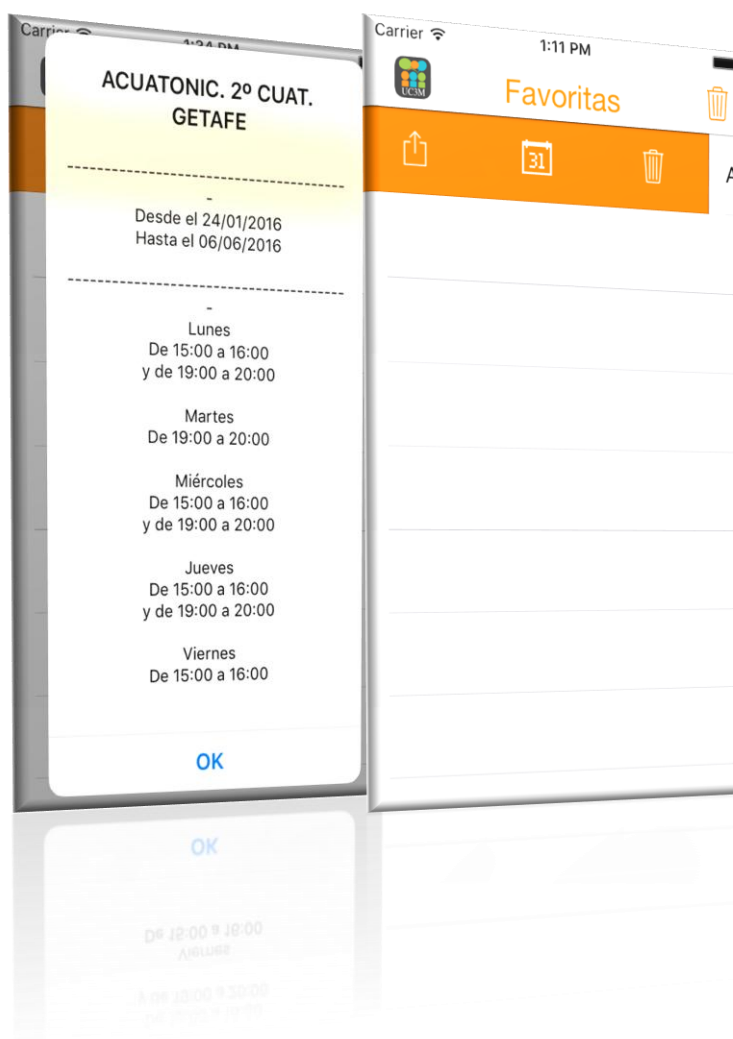
Nuestra tercera imagen muestra como se verían los horarios en la ventana emergente, al igual que en Actividades y en Mis Actividades el pop-up de horarios es idéntico.

Para concluir con la vista Favoritas, analizaremos el elemento más característico y que se encuentra un poco oculto y son como se puede apreciar en la segunda imagen, las celdas deslizantes con opciones. Se trata de celdas que pueden deslizarse y albergan en su interior una serie de funcionalidades:

- Compartir: cómo podemos ver en la vista Favoritas no había botón de compartir, había quedado anulado por la papelera de borrar todas las actividades favoritas, con este detalle podemos compartir la actividad que deseemos.
- Horario: se trata de un acceso directo, no hace falta abrir la vista en

detalle para poder ver los horarios, con esta opción podemos ojear de manera rápida y sencilla las fechas y horas en que se imparte una actividad.

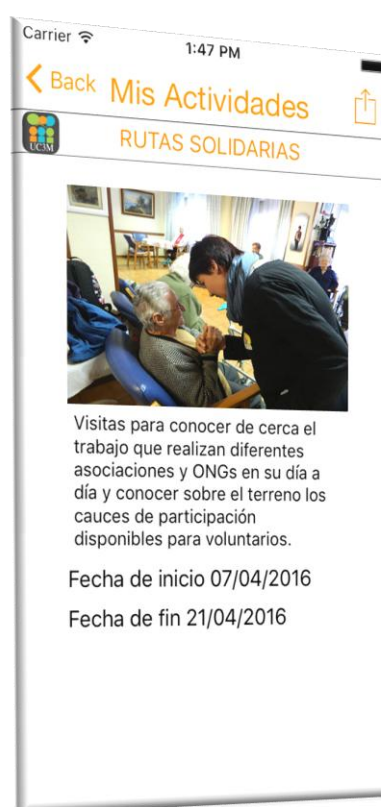
- Borrar: es la versión simplificada del borrar genérico, con este botón papelera podemos borrar únicamente la actividad seleccionada.





**Mis Actividades** es probablemente la función menos utilizada de la aplicación, por lo general los usuarios se inscriben en un par de actividades y suelen tener claro horarios y todo tipo de dudas sobre la actividad, pero para aquellos usuarios más despistados siempre podrán comprobar la información de las actividades inscritas en este apartado de la aplicación, además también sirve como seguro de haber hecho bien los trámites de inscripción aunque a veces desde el periodo en que un usuario se inscribe hasta que aparece como inscrito no es inmediato y esto puede dar a confusión al cliente.

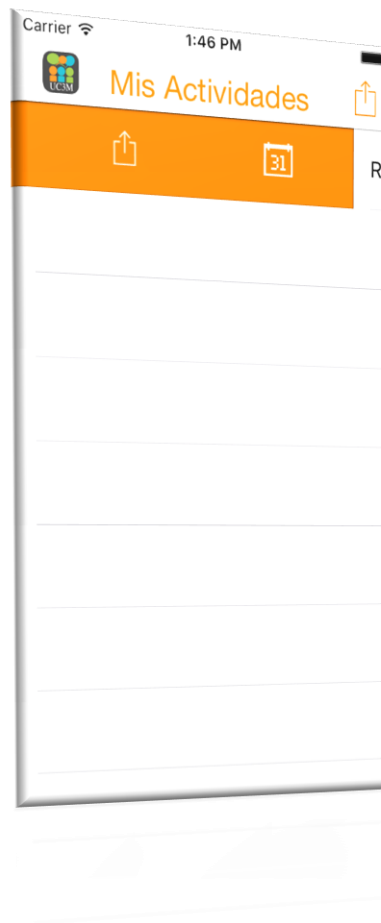
Básicamente es una copia de Favoritas pero más sencilla y con menos posibilidades, si en Favoritas podíamos interactuar mucho más con la aplicación, Mis Actividades es prácticamente una vista informativa, se pierden elementos como:



**Ilustración 29: Mis Actividades 1.**

- Botones de borrar: no se puede borrar una aplicación inscrita, si el usuario deseara darse de baja, debería acudir a una oficina de ESES o contactar por algún medio telemático.

- Botón de inscribir: evidentemente no se puede inscribir en una actividad inscrita.
- Botón de compartir: está duplicado pero, se prefiere dejar el general ya que en toda nuestra aplicación estará en todas las vistas y debemos de seguir con el estándar, el botón de compartir de la celda deslizante podría eliminarse por redundancia, es una reseña que se estudiará en versiones posteriores.
- Botón de borrar: se está estudiando eliminar las celdas deslizantes y añadir la opción de ver horario en la vista de detalle, pero la usabilidad de la aplicación demanda hacer el menor número de transiciones para poder activar una funcionalidad, hace más claro al usuario localizarse dentro de la aplicación. Como en el caso anterior se estudiará hacer modificaciones en las versiones posteriores, depende mucho del análisis sobre el uso de la aplicación.



**Ilustración 30: Mis Actividades 2.**



Las Vistas de promociones son muy parecidas a las vistas de Actividades, al igual que Mis Actividades con Favoritas se ha reutilizado la idea y hemos utilizado el carrusel como forma de visualizar las promociones. Además se incluyen las pestañas de ISIC-ITIC y Universidad, que diferencian entre el personal Estudiante (Student) y el personal Profesorado (Teacher) con Tarjetas TE de los estudiantes estándar que no poseen la tarjeta TE.

Como se puede observar en las imágenes, se han eliminado muchos elementos correspondientes a Actividades y se han introducido otros nuevos.

Como se puede apreciar en las imágenes, la vista pierde elementos y gana en claridad y limpieza ya dijimos con anterioridad que la vista Actividades era la más funcional a la vez que la más caótica por su cantidad de elementos funcionales e informativos. Se mantienen título e imagen y se pierden elementos como el video o los precios, se añaden dos elementos característicos de esta vista:



**Ilustración 31: Promociones 1.**

- Fecha Fin: define la fecha en la que caduca la promoción, si una promoción ha caducado no se mostrará en el carrusel, hay casos en los que no existe fecha de caducidad conocida (promociones por temporada), en este caso no se define fecha, y se muestra “No definida”.
- Enlace externo: Al igual que la fecha, hay ocasiones en las que no hay un enlace externo, en este caso no aparecerá ningún enlace y se deduce que no existe, cuando existe un enlace externo se muestra como en la red (letras azules y moradas al ser pulsadas) al ser pulsado la aplicación abrirá un navegador del dispositivo y nos redirigirá a la dirección del enlace de la promoción, esta opción nos permite ver en detalle las condiciones de la promoción.



Ilustración 32: Promociones 2.

Quizás esta sea una de las vistas más llamativas para el usuario puesto que utiliza recursos del dispositivo como las llamadas, las redes sociales y la geo-localización.

La vista principal se divide en dos partes claramente diferenciadas, la parte superior y la parte central-baja.

Superior: Aglutina las posibles funciones para contactar vía internet, se da la posibilidad de contactar por:

- **Twitter** a la cuenta oficial de ESES.
- **Facebook** poder contactar con ESES en su muro principal o mediante mensaje si somos amigos.
- **Correo**, se abrirá nuestro gestor de correo por defecto con la dirección de ESES pre escrita para facilitar el envío.



**Ilustración 33: Contacto.**

Como se comentó con anterioridad la funcionalidad de estas vistas sin acceso a la red es nula.

Central-Baja: Consta de un nuevo elemento en la aplicación, se trata de la ruleta, es conocida por utilizarse para marcar fechas, pero en nuestro caso se ha utilizado para Mostrar los campus que tienen una oficina física de Espacio Estudiantes. El usuario debería de seleccionar un campus y después utilizar una de las dos opciones que hay en la parte baja:

- **Llamada:** al pulsar la aplicación comenzará una llamada al teléfono de la oficina marcada en la ruleta, cabe destacar que si no tenemos cobertura de llamada, el sistema no dejará realizar la llamada.
- **Localización:** al pulsar se transita a una nueva vista, que contiene un mapa de Google Maps, se comento con anterioridad el uso de este Framework para gozar de la posibilidad de ver los interiores de los edificios y poder marcar mejor el aula exacta en que se encuentra ESES así como la planta del edificio.

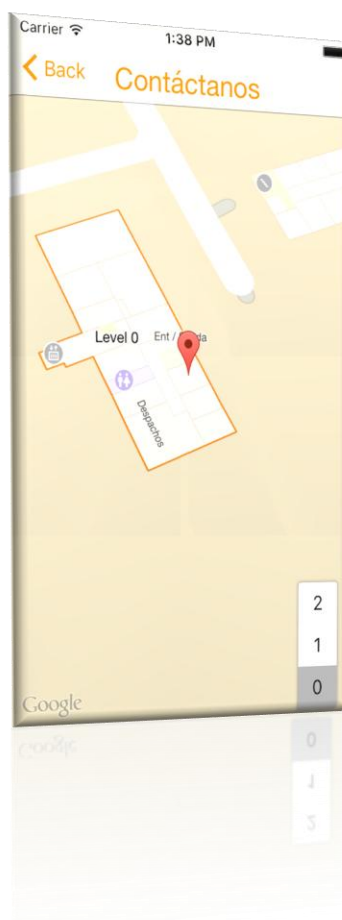
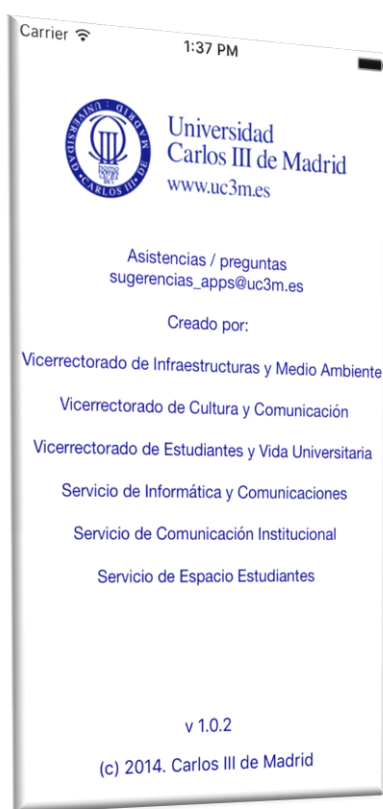


Ilustración 34: Mapas.

Pasaremos ahora a comentar una de las vista menos utilizadas y con menos funcionalidad de la aplicación, la vista de información que tiene un pequeño detalle único en el resto de vistas.

Esta vista como se puede apreciar en la imagen muestra los departamentos encargados de realizar la aplicación así como el Logotipo corporativo de la universidad Carlos III de Madrid, el año de publicación de la aplicación y quizás uno de los elementos más importantes de la vista, el número de versión en que se encuentra nuestro cliente.

Como se puede ver, pierde la estética estándar de la aplicación y no hay botones, ni desplegados, ni pop-ups, ni carrusel, por no tener, no tiene ni botón de retorno como en muchas vistas, esto tiene una sencilla explicación, la vista Información es la raíz de la rama información y no puede volver atrás porque solo se accede desde el menú lateral y este está superpuesto al resto y no es la vista padre de ninguna de las raíces de navegación.

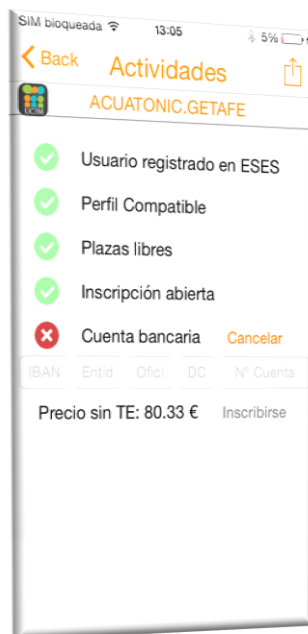


Pero a pesar de todo esto el menú que siempre habíamos desplegado mediante la pulsación en un botón con el icono de la aplicación situado en la esquina superior izquierda de todas las vistas, ahora no aparece sin embargo el menú drawer tiene una característica muy peculiar que se puede utilizar en todas las vista y es su emergencia mediante un suave desliz del borde izquierdo de la pantalla hacia el borde derecho de la misma, el menú está escondido y emerge mediante un evento de desliz. Pasa algo parecido a las celdas deslizantes de las Vistas Favoritas y Mis Actividades, pueden pasar desapercibidas para algunos usuarios.

**Ilustración 35: Información.**

Llegamos ahora a una de las vistas más especiales de la aplicación, la hemos dejado para el final porque se puede acceder a ella desde varias ramas de la navegación interna de la app y no queríamos inducir a confusión, como recordaremos las vistas de Actividades y la vista en detalle de una actividad Favorita permitía mediante un botón transitar a una nueva vista hasta entonces fantasma llamada inscripción, pues bien es el momento de analizar nuestra vista de inscripción que tiene como función principal controlar que el usuario cumple los requisitos básicos para su inscripción.

La vista realiza una petición al servidor y recibe los datos mostrando por pantalla si el usuario cumple todos los requisitos, si todo es correcto el botón de inscribirse se desbloquea y podemos realizar la inscripción, si falta el dato de cuenta bancaria podemos rellenarlo en ese mismo momento y cumplir con todos los requisitos, destacar que los campos de texto de la cuenta bancaria tienen una gran cantidad de restricciones y se adaptan perfectamente a un número de cuenta bancaria, además el botón de inscribirse solo se desbloqueará si todos los campos son correctos por lo que no hay riesgo de hacer una inscripción fallida con datos erróneos.

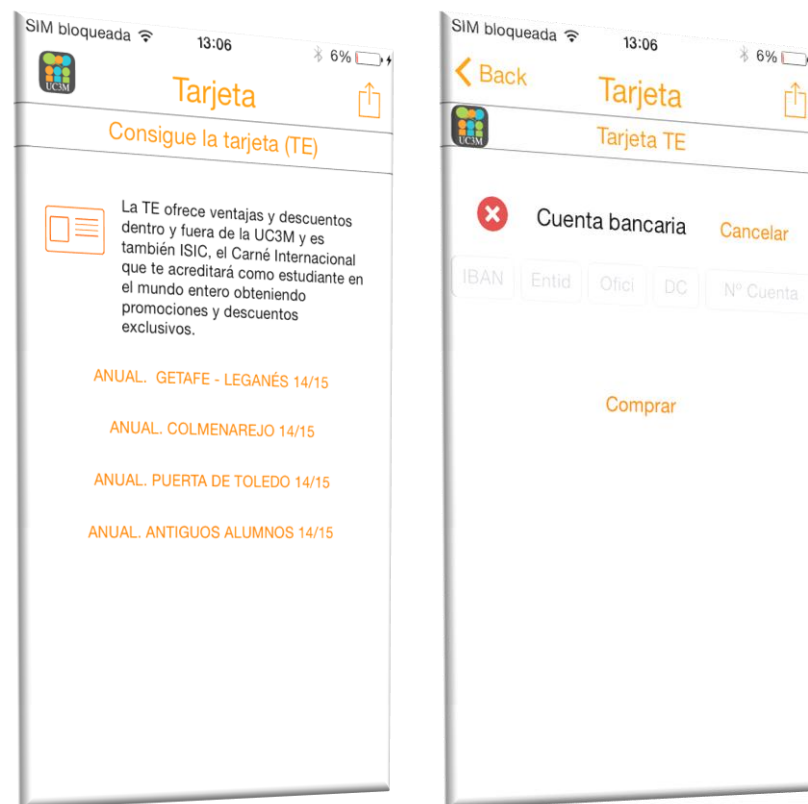


En próximas versiones la aplicación madurará y permitirá realizar cambios en otros aspectos de los requisitos para inscribirse pero actualmente el único campo que permite realizar cambios es el número de cuenta bancaria, si se desea cumplir con cualquiera de los demás aspectos se debe de hacer mediante las oficinas de Espacio Estudiantes o telemáticamente en la versión web.

**Ilustración 36: Inscripción.**

Para finalizar con las interfaces de usuario de la aplicación vamos a describir la vista de Tarjetas, la hemos dejado para el final por tener un cierto parecido a la vista de inscripción y así comprender mejor su diseño.

Básicamente se compone de dos vistas, la primera de ellas muestra las tarjetas que puede obtener un usuario, estos tipos de tarjetas son variables y no siempre se puede optar a todas, evidentemente si el usuario ya posee una tarjeta, esta dejará de aparecer como posible compra. No tendría lógica ofrecer una tarjeta a un usuario que ya posee dicha tarjeta.



**Ilustración 37: Tarjetas.**

La vista principal también muestra una pequeña descripción de la tarjeta y sus beneficios, esta descripción es única e igual para todas las tarjetas, una vez seleccionamos el tipo de tarjeta que queremos, se realiza una transición hacia la segunda vista que se asemeja bastante a la vista de inscripción, evidentemente podemos observar como muchos elementos han desaparecido debido a que los requisitos para comprar una tarjeta con mínimos, ser usuario de la UC3M,



teniendo acceso a esta vista está claro que es un requisito que se cumple y tener un número de cuenta bancaria disponible para poder realizar el cobro de la compra.

Desaparecen el resto de requisitos y destacar que el botón comprar también cumple con la seguridad de estar activo únicamente cuando hay un número de cuenta disponible, también se permite realizar el cambio del número de cuenta por si el usuario desea hacerlo desde otra, esto es peligroso en un sentido, si un usuario realiza el cambio, este se aplicara a la base de datos y ese nuevo número de cuenta será el oficial del alumno para los cobros de ESES.

Con esto concluimos las vistas que componen nuestra aplicación y la funcionalidad que realizan cada una de ellas.



#### 4.4 DIAGRAMA DE NAVEGACIÓN.

El siguiente diagrama muestra como las vistas transitan de una a otra. Mencionar como se dijo anteriormente que la vista menú es una vista superpuesta a las demás y no es la raíz de ningún árbol de navegación aunque pueda parecer lo contrario, por lo tanto cada una de las opciones del menú crea su propio árbol de navegación que en nuestro caso se trataría de un árbol de una sola rama, por ello la vista información no tiene botón de regreso puesto que sería la raíz de su árbol de una sola rama y una sola hoja.

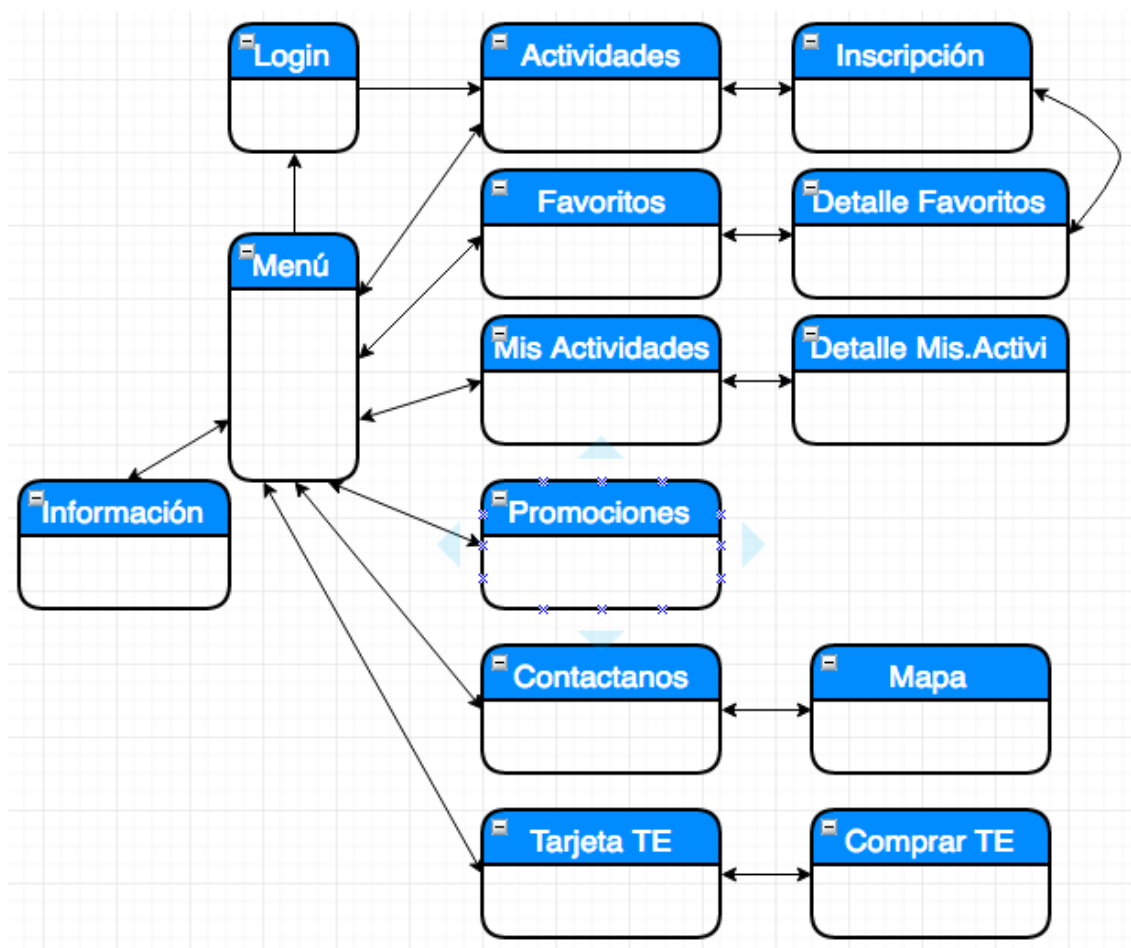


Ilustración 38: Diagrama de navegación.

Otra característica a destacar es que a pesar de que Detalle Favoritas y Actividades pueden transitar a la misma vista Inscripción, no está permitido transitar desde Detalle Favorito hasta Actividades pasando por inscripción ni viceversa, la única manera de hacer esto es ir al menú y cambiar de rama.

Por último denotar que aunque se recalcó con anterioridad, menú es una vista superpuesta y se puede acceder a ella desde cualquier hoja de cualquier rama, excepto desde la vista login con lo cual debería de haber una flecha que uniese



cada vista con la vista menú, se ha obviado para que se vea claramente cuál es el orden de navegación lógico.



## CAPÍTULO 5: IMPLEMENTACIÓN Y PRUEBAS.

La implementación es la parte del proyecto que más duración conlleva y su resultado es la propia aplicación, en este paso se materializa todo lo descrito anteriormente en el documento, por ello vamos a analizar parte de esta implementación, sobre todo de la parte de comunicación entre los servicios web que es donde surge la magia entra las aplicaciones cliente y los datos de la SGBD.

Además no hay que olvidarse de realizar la fase de pruebas pues estas nos ayudan a detectar errores tempranos y a corregirlos antes de que sean más complicados de solucionar, esta fase nos da la seguridad de que la aplicación es de calidad y funcionará correctamente en los dispositivos del cliente.

## 5.1. IMPLEMENTACIÓN.

A continuación mostraremos un ejemplo de cómo se comporta una llamada desde el servidor de comunicaciones.

```
@Getter
@Setter
@ToString(of = {"codigo", "nombre", "inicio"})
@EqualsAndHashCode(of = "codigo")
public class Actividad implements Comparable<Actividad>, Pagable {

    //Atributos
    private String codigo;
    private String nombre;
    private String descripcion;
    private Campus campus;
    private TipoActividad tipo;
    private String cuentaContable;
    private boolean listaEsperaPermitida;
    private Integer inscritos;
    private Integer enEspera;
    private String URLImagen;
    private String URLVideo;
    private SortedSet<DiaHora> horarios = new TreeSet<DiaHora>();

    //Métodos
    public boolean isInscripcionAbierta() {
    }
    public boolean isPlazasLibresDisponibles() {
    }
    public Float getPrecioBase() {
    }
    public Float getPrecioTarjetaESES() {
    }
}
```

Gracias a Lombok una librería de java o POJO que podemos añadir como dependencia a nuestro proyecto, nos olvidaremos de realizar todos los getters y setters de nuestras clases, ¿Qué quiere decir esto? Que nos ahorramos la implementación de los métodos así como de sus constructores y de otros métodos comunes como puedan ser toString, de esta manera tendremos un código limpio y con acceso y modificación de cualquier parámetro de la clase.

```
import es.uc3m.adys.eses.Aplicacion;
import es.uc3m.adys.eses.excepciones.UsuarioException;
import es.uc3m.adys.eses.modelo.Actividad;
import es.uc3m.adys.eses.modelo.Campus;
import es.uc3m.adys.eses.modelo.RequisitosInscripcion;
import es.uc3m.adys.eses.servicio.EsesServicio;
import net.fortuna.ical4j.model.*;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

```
import java.io.IOException;
import java.util.*;

@RestController
@RequestMapping("/v1/actividades")
public class ActividadController {
    @Autowired
    private EsesServicio svc;

    @RequestMapping(method = RequestMethod.GET)
    public Collection<Actividad> getActividadesPorCampus(HttpServletRequest request re
q, HttpServletResponse res, @RequestParam(required = true, value = "campus") I
nteger codCampus) throws IOException {
        final Campus campus = svc.findCampusByCodigo(codCampus);
        Long idu = (Long) req.getAttribute(Aplicacion.UC3M_IDU);
        try {
            return campus != null ? svc.findAllActividadesByCampus(campus, idu
) : new ArrayList<Actividad>();
        }
        catch (UsuarioException ue) {
            res.sendError(HttpServletResponse.SC_INTERNAL_SERVER_ERROR, ue.get
Message());
            return null;
        }
    }

    @RequestMapping(method = RequestMethod.GET, value = "{codActividad}")
    public Actividad getActividad(@PathVariable(value = "codActividad") String
codActividad) {
        return svc.findActividadByCodigo(codActividad);
    }
}
```

El siguiente código muestra un ejemplo de controlador que como podemos observar utiliza la clase Actividad, la magia se desarrolla con las anotaciones @RequestMapping que asigna la URL o dirección de acceso al servicio, esta URL es la dirección que debe de utilizar nuestro cliente para realizar las lecturas, escrituras y modificaciones en la base de datos y cómo podemos observar está controlado por el controlador, así que solo podrá realizar las acciones pertinentes que el controlador le permita.

Como podemos observar a esta URL le podemos asignar terminaciones que añaden servicios, como en el ejemplo codActividad que se añade a /actividades y presenta la actividad solicitada mediante el código.

También podemos apreciar que este método realiza una llamada a findActividadByCodigo, este método aunque lleve notación estándar, está creado por el programador y será nuestro próximo punto de análisis.

```
import es.uc3m.adys.eses.modelo.*;
import es.uc3m.adys.eses.persistencia.*;
import es.uc3m.datos.centros.Centro;
import es.uc3m.datos.personal.Relacion;
import es.uc3m.datos.personas.Persona;
import lombok.extern.slf4j.Slf4j;
import org.apache.commons.validator.routines.checkdigit.IBANCheckDigit;
import org.joda.time.DateTimeConstants;
```

```
import org.joda.time.LocalDate;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.cache.annotation.Cacheable;
import org.springframework.stereotype.Component;
import org.springframework.stereotype.Service;
import org.springframework.transaction.annotation.Transactional;

@Service
@Component
@Slf4j
public class EsesServicio {

    @Autowired
    private CampusMapper campusMapper;

    @Autowired
    private ActividadMapper actividadMapper;

    @Autowired
    private PersonaDeporwinMapper personaDeporwinMapper;

    @Cacheable(value = "CAMPUS")
    public List<Campus> findAllCampus() {
        return campusMapper.findAll();
    }

    public Campus findCampusByCodigo(Integer codCampus) {
        for (Campus c : findAllCampus())
            if (c.getCodigo().equals(codCampus))
                return c;
        return null;
    }

    @Cacheable(value = "ACTIVIDADES")
    public SortedSet<Actividad> findAllActividadesByCampus(final Campus campus
, Long idu) throws UsuarioException {
        Usuario u = findDatosUsuario(idu);
        if (u != null && u.getIdPersonaDeporwin() != null) {
            return actividadMapper.findActividadesByCampus(campus.getCodigo(),
u.getIdPersonaDeporwin());
        } else
            throw new UsuarioException("Usuario con idu `" + idu + "`no encont
rado en Deporwin");
    }

    @Cacheable(value = "ACTIVIDADES")
    public Actividad findActividadByCodigo(String c) {
        return actividadMapper.findActividadByCodigo(c);
    }
}
```

Este fragmento de código es el servicio de llamadas asignado a las Actividades, el cliente hace la llamada a la URL oportuna esta llamada recibe respuesta del controlador y el controlador aplica el servicio específico de la llamada, como se puede ver en el método `findActividadesByCodigo` que es el que estamos utilizando de ejemplo, se hace una llamada a `findActividadByCodigo` y aunque el

nombre sea igual son métodos de clases diferentes, nuestro controlador llama al primer `findActividadesByCodigo` que pertenece a la clase `EsesServicio` y una vez dentro de la clase `EsesServicio` hay un método que tiene que coincidir con dicho nombre, este a su vez realiza una llamada a otro método con el mismo nombre pero esta vez se encuentra en `actividadMapper` que es nuestro último paso para acceder a la base de datos.

```
<mapper namespace="es.uc3m.adys.eses.persistencia.ActividadMapper">

    <sql id="queryBase">
        select  a.codigo,
                a.descripcion as nombre,
                ta.Codigo as cod_tipo,
                a.SubTipoActividad as cod_campus,
                ae.Imagenes as url_imagen,
                ae.Enlaces as url_video,
                a.descuentoAbonado as dto_abonado,
                a.tipoDescuentoAbonado as tipo_dto_abonado,
        from actividades a
            left join ActividadesDatosExtra ae on
                ae.CodigoActividad=a.Codigo
            inner join TiposActividad ta on
                ta.Codigo=a.TipoActividad
    </sql>

    <select id="findActividadByCodigo" parameterType="map" resultType="es.uc3m
.adys.eses.modelo.Actividad" resultMap="resActividad">
        <include refid="queryBase" />
        Where a.codigo=#{codActividad}
    </select>

    <resultMap id="resActividad" type="es.uc3m.adys.eses.modelo.Actividad">
        <id property="codigo" column="codigo"/>
        <result property="nombre" column="nombre"/>
        <result property="descripcion" column="descripcion"/>
        <result property="idAgrupacion" column="idAgrupacion"/>
        <result property="cuentaContable" column="cuentaContable"/>
        <result property="urlImagen" column="url_imagen"/>
        <result property="urlVideo" column="url_video"/>
        <association property="campus" javaType="es.uc3m.adys.eses.modelo.Campus">
            <id property="codigo" column="cod_campus"/>
            <result property="nombre" column="campus"/>
        </association>
    </resultMap>

</mapper>
```

El siguiente código es el último de nuestro servicio web y es donde se relacionan las clases de java con los objetos de la base de datos, aquí se definen las consultas y los tipos de datos java que devuelven de tal forma que si una consulta devuelve una actividad con 5 campos String, debemos de tener una



clase con 5 atributos String que pueda almacenar los datos de la consulta temporalmente en un objeto de dicha clase que devolverá al cliente.

Por lo tanto el cliente hará una llamada al Web Service desde una URL permitida por el Web Service, este Web Service recibe la llamada en su controlador y crea los objetos de las clases necesarias, estos objetos serán rellenos con los datos procedentes de la base de datos, para recibir estos datos, el controlador llama al servicio que es el encargado de comunicar con el fichero de mapeo entre clases java y objetos de la base de datos, cuando el objeto de la clase java está completo se envía al cliente mediante JSON, para que se puedan tratar los datos de una manera fácil y sencilla.

## 5.2 PRUEBAS.

Para la correcta certificación de calidad del producto se han realizado una batería de pruebas que verifican el correcto funcionamiento de la aplicación antes de ofrecer el producto al usuario. A continuación se muestra y detalla el modelo de tabla que especifica cada una de las pruebas.

**Identificador: <tipo de prueba> - <número>**

Nombre:	
Descripción:	
Necesidad:	
Prioridad:	
Fuente:	

**Tabla 117: Descripción Tabla de Pruebas funcionales.**

Los campos se rellenarán en función de la siguiente definición:

- **Identificador:** Código único que identifica de manera unívoca a cada prueba. Consta de tres partes: tipo de prueba, subtipo de prueba y número.
- **Tipo de prueba:** Existen dos tipos de prueba: pruebas funcionales(PF) y pruebas de estrés (PE).
- **Número:** Cada vez que se cambie de tipo, la numeración de los requisitos volverá a comenzar en 1.
- **Nombre:** Título que identifica la prueba.
- **Descripción:** Definición sobre la prueba.
- **Necesidad:** Grado en el que la prueba debe ser satisfecha. Puede tomar los valores de “Muy Alto”, “Alto”, “Medio”, “Bajo” y “Muy Bajo”.
- **Prioridad:** Nivel de urgencia con el que debe cumplirse la prueba. Puede tomar los valores de “Muy Alto”, “Alto”, “Medio”, “Bajo” y “Muy Bajo”.
- **Fuente:** Indica la procedencia de la prueba, pudiendo proceder de documentos o personas.

### 5.2.1 PRUEBAS FUNCIONALES.

A continuación se indicará cada una de las pruebas funcionales realizadas acompañadas de capturas de la aplicación en ejecución.

Las primeras pruebas se centran sobre la primera vista que carga la aplicación al ejecutarse, la vista de Login.

Esta vista consta de 2 elementos problemáticos, con alta frecuencia de generar errores o comportamientos no deseados.

El primero de ellos es el elemento Textfield o campo de texto, donde el usuario deberá rellenarlos con sus datos de acceso.



El segundo de ellos es el elemento Label de alerta, que parpadea al detectar un acceso incorrecto indicando el error cometido.

Además cabe destacar que todos los errores son generados por el evento “pulsar botón de Inicio” que verifica los campos de texto y nos permite acceder a los servicios de la aplicación.

**Ilustración 39: Prueba login.**

### Identificador: PF-01

Nombre:	Usuario vacío
Descripción:	Comprobación del campo de texto Usuario, no puede estar vacío en el momento de acceder a la aplicación.
Necesidad:	Media.
Prioridad:	Alta.
Fuente:	Encargado de Seguridad, Encargado de pruebas, Analista..

**Tabla 118: Prueba Funcional 1.**



Como se puede apreciar en la ilustración 40, el Label de información parpadea mostrando un mensaje de error indicando “Falta el nombre de usuario”.

El Label desaparece al cabo de unos segundos y la vista vuelve a su estado original.

El Label de información parpadea en color rojo para dar más énfasis al error, el rojo siempre es un tono asociado a prohibición o error.



**Ilustración 40: Prueba login 1.**

## Identificador: PF-02

Nombre:	Contraseña vacía.
Descripción:	Comprobación del campo de texto Contraseña, no puede estar vacío en el momento de acceder a la aplicación.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Encargado de pruebas, Analista.

**Tabla 119: Prueba Funcional 2.**



Como se puede apreciar en la ilustración 41, el label de información parpadea mostrando un mensaje de error indicando “Falta la clave”.

El label desaparece al cabo de unos segundos y la vista vuelve a su estado original.

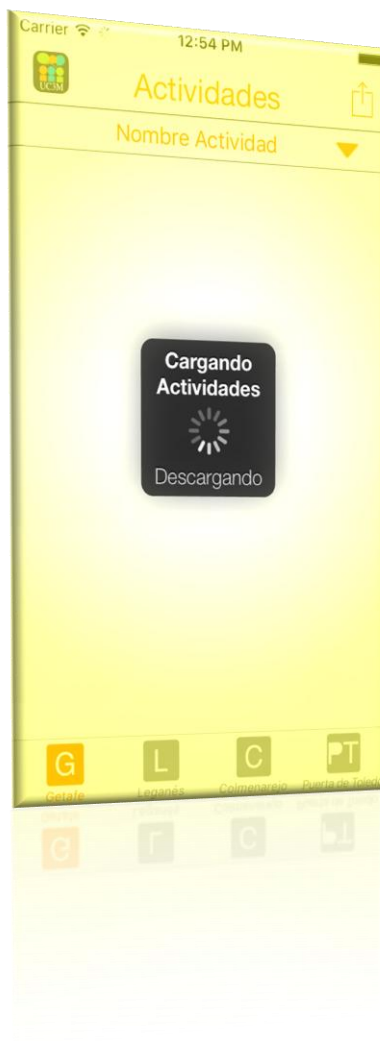


**Ilustración 41: Prueba login 2.**

### Identificador: PF-03

Nombre:	HUD y botón de recarga.
Descripción:	Comprobación del correcto funcionamiento del HUD y del botón de recarga cuando hay desconexión.
Necesidad:	Baja.
Prioridad:	Media.
Fuente:	Encargado de pruebas, Analista.

**Tabla 120: Prueba Funcional 3.**



Como se puede apreciar en la ilustración 42, el usuario está dentro de la aplicación y podrá observar las actividades del campus de Getafe que siempre es la primera vista al entrar.

Antes siempre aparece un HUD de carga por si sucede alguna desconexión de red.

**Ilustración 42: Prueba HUD y botón de recarga.**

## Identificador: PF-04

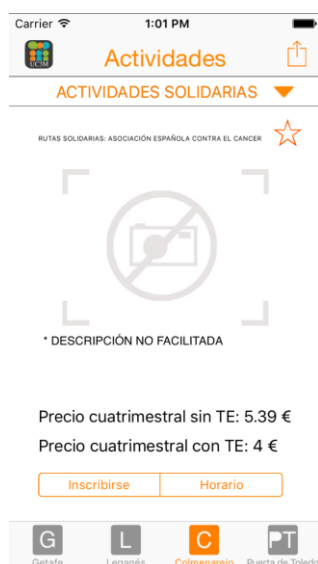
Nombre:	Imagen correcta.
Descripción:	Comprobación del campo imagen en la vista de Actividades.
Necesidad:	Media.
Prioridad:	Alta.
Fuente:	Encargado de pruebas, Analista.

Tabla 121: Prueba Funcional 4.

Cuando el sistema carga las actividades de un campus instancia 3 vistas de actividad y carga 3 imágenes, cuando movemos el carrusel adelante o atrás, el sistema cambia los valores de la imagen de las 3 instancias y si movemos el carrusel muy rápido pueden surgir problemas de carga. También puede producirse un micro corte de conexión o sencillamente que el dato venga erróneo en cuyo caso se carga la siguiente imagen:



Si por alguna razón no hay imagen de actividad se cargará la siguiente imagen:



(No existe imagen para esta actividad). (No se ha podido cargar la imagen).

## Identificador: PF-05

Nombre:	Estrella Favoritas.
Descripción:	Comprobación del botón Favoritas en la vista de Actividades.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Encargado de pruebas, Analista.

**Tabla 122: Prueba Funcional 5.**

En la vista Actividades, comprobar que funciona correctamente el botón Favoritas, pulsamos y cambia de color, volvemos a pulsar y vuelve a cambiar de color. También se ha probado mover el carrusel y comprobar que las favoritas permanecían marcadas, en una fase del desarrollo al marcar un favorito, este se replicaba cada 3 vistas debido a que al haber tres instancias de la vista actividad, no se modificaba el parámetro, está reparado.



**Ilustración 43: Estrella favorito desmarcada.**



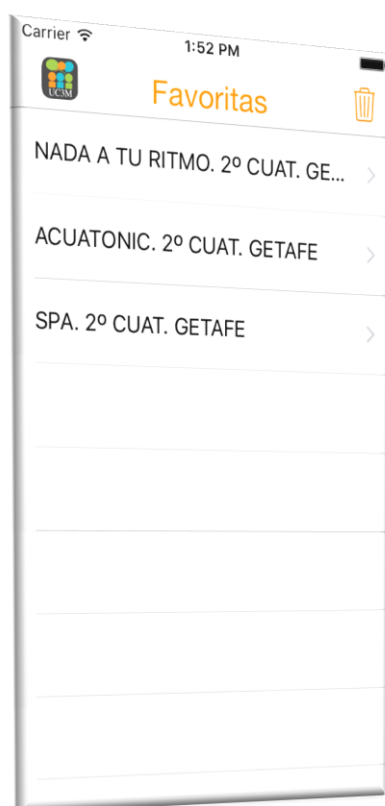
**Ilustración 44: Estrella favorito marcada.**



## Identificador: PF-06

Nombre:	Favorito queda registrado en Favoritas.
Descripción:	Comprobación de la actividad marcada como favorita aparece en la vista Favoritas.
Necesidad:	Muy Alta.
Prioridad:	Alta.
Fuente:	Encargado de pruebas, Analista.

**Tabla 123: Prueba Funcional 6.**



En la vista Favoritas podemos observar como se ha agregado correctamente la celda de la actividad favorita marcada en la vista de Actividades.

También comprobamos que al borrar un favorito de la vista Favoritas se borra tanto de la vista Favoritas como de la estrella de la actividad en la vista Actividades.

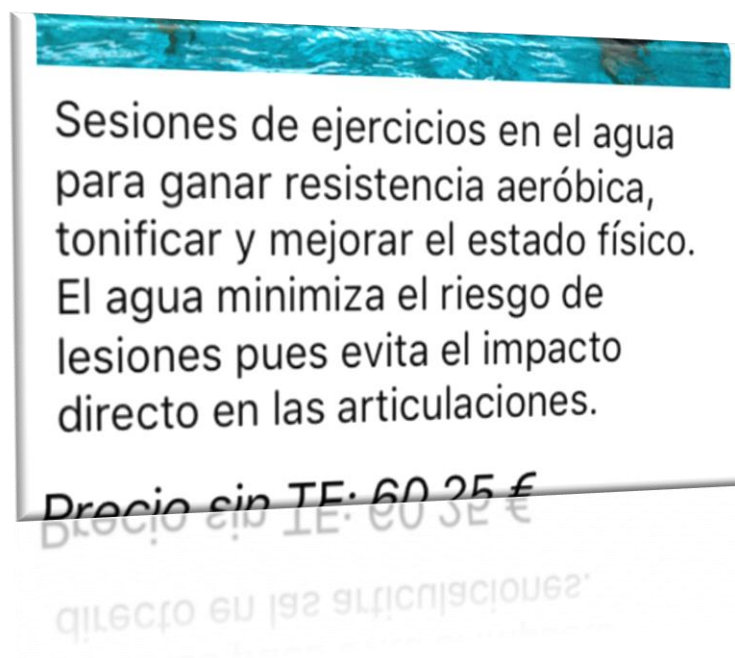
**Ilustración 45: Prueba Favoritos.**

### Identificador: PF-07

Nombre:	Scroll descripción, nombre y precios.
Descripción:	Comprobación que el Scroll de las descripciones funciona correctamente, así como la visualización del nombre y de los precios.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Encargado de pruebas, Analista.

**Tabla 124: Prueba Funcional 7.**

Esta prueba se aplica a todas las vistas que tienen descripción de actividad como puedan ser las vistas de Actividades, Detalle de Favorito, Detalle de Actividad inscrita o Promociones.

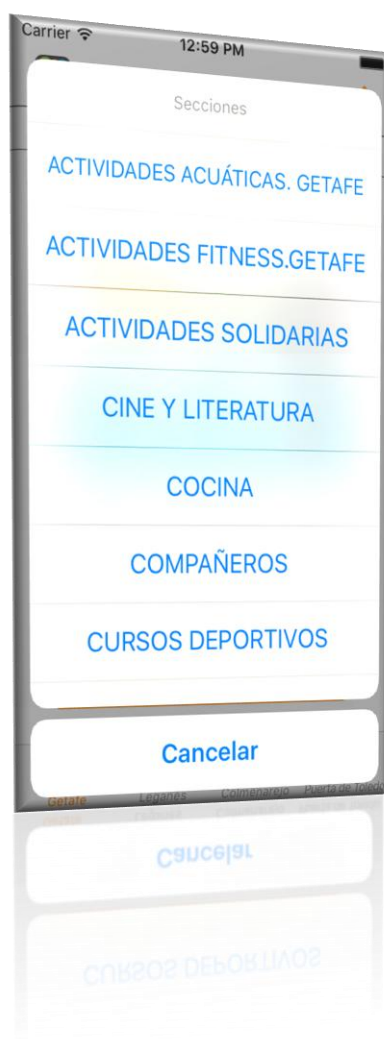


**Ilustración 46: Prueba descripción, nombre y precios.**

## Identificador: PF-08

Nombre:	Desplegable sección.
Descripción:	Comprobación del correcto funcionamiento de las secciones.
Necesidad:	Media.
Prioridad:	Media.
Fuente:	Encargado de pruebas, Analista.

**Tabla 125: Prueba Funcional 8.**



Esta prueba se aplica al desplegable de sección en la vista Actividades, se pulsa el desplegable se comprueba que se despliega correctamente y que está cargado con los tipos de actividades, luego se pulsa sobre una de las secciones y verificamos el correcto funcionamiento del carrusel que nos traslada hasta la primera actividad de la sección indicada de manera automática.



**Ilustración 47: Prueba tipos de actividad.**

## Identificador: PF-09

Nombre:	Desplegable Horario.
Descripción:	Comprobación del correcto funcionamiento de los horarios.
Necesidad:	Alta.
Prioridad:	Alta.
Fuente:	Encargado de pruebas, Analista.

Tabla 126: Prueba Funcional 9.



Ilustración 48: Prueba Horario.

Esta prueba se aplica al desplegable de horario en la vista Actividades, se pulsa el botón horarios y se comprueba que se despliega correctamente y que está cargado con los horarios pertinentes a dicha actividad.

Esta prueba se ha realizado para todas las vistas que contienen horario, como puede ser Actividades, Favoritas, Detalle Favorito o Mis Actividades.

## Identificador: PF-10

Nombre:	Carrusel, botón de reproducir y precios.
Descripción:	Comprobación del correcto funcionamiento del botón reproducir video de la actividad y desliz del carrusel.
Necesidad:	Baja.
Prioridad:	Media.
Fuente:	Encargado de pruebas, Analista.

**Tabla 127: Prueba Funcional 10.**



Esta prueba se aplica a las vistas Actividades, Detalle Favorito y Detalle Mis Actividades. Se comprueba que el botón a aparece cuando la actividad tienen URL del video, en caso contrario solo aparece la imagen de la actividad.

Se pulsa sobre el botón y se comprueba el correcto funcionamiento del reproductor multimedia.

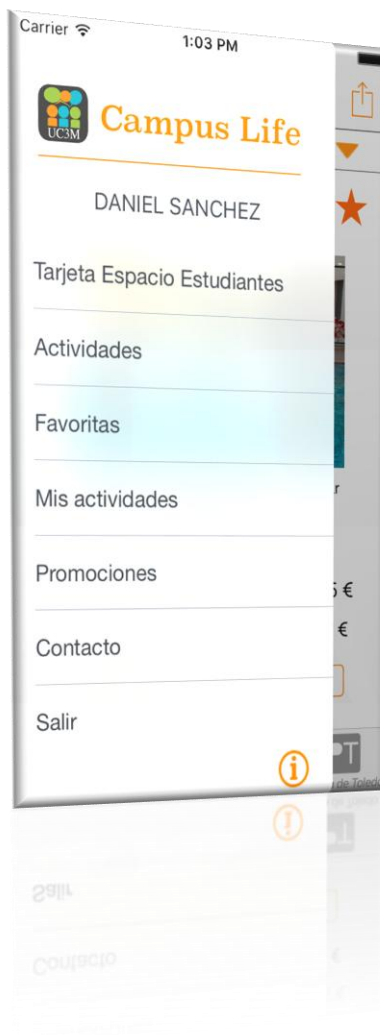
Se comprueba el precio cuando es 0 y el correcto funcionamiento del carrusel, debe de ser suave y preciso.

**Ilustración 49: Prueba Actividades.**

## Identificador: PF-11

Nombre:	Menú Drawer.
Descripción:	Comprobación del correcto funcionamiento del menú lateral.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de pruebas, Analista.

**Tabla 128: Prueba Funcional 11.**



**Ilustración 50: Prueba Menú.**

Esta prueba se aplica a todas las vistas de la aplicación exceptuando la vista Login. Todas y cada una de las vistas pueden acceder al menú tanto por botón como por desliz lateral, excepto la vista información que solo puede acceder mediante desliz lateral.

Se comprueba también la correcta transición a las vistas deseadas y el correcto cierre de la aplicación desde el botón de salir.

Además se prueba que los datos de usuario son efectivamente los correspondientes a la persona que se está logueando a la aplicación.

## Identificador: PF-12

Nombre:	Pestañas.
Descripción:	Comprobación del correcto funcionamiento de las pestañas.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de pruebas, Analista.

**Tabla 129: Prueba Funcional 12.**

Esta prueba se aplica en las vistas de Actividades y en la vista Promociones. Probamos la correcta carga de actividades y promociones respectivamente tras realizar un cambio de pestaña y comprobamos que al volver a la pestaña anterior, sigue cargada con los mismos datos que cuando se abandonó.

Al cambiar de raíz en la navegación se pierden las instancias y hay que volver a cargar de nuevo toda la información.

Estos son los botones que hay que probar en Actividades, en Promociones se trata de texto simple.



### Identificador: PF-13

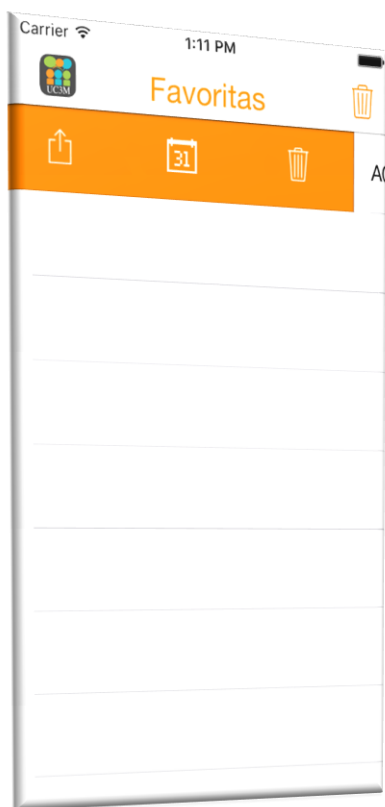
Nombre:	Celdas deslizantes y menú oculto.
Descripción:	Comprobación del correcto funcionamiento de las celdas deslizantes y las opciones situadas debajo.
Necesidad:	Media.
Prioridad:	Alta.
Fuente:	Encargado de pruebas, Analista.

**Tabla 130: Prueba Funcional 13.**

Esta prueba se aplica en las vistas de Favoritas y en la vista Mis Actividades.

Probamos como deslizando el dedo de izquierda a derecha las celdas muestran lo que hay bajo de ellas, es importante probar la viabilidad con el menú deslizante puesto que el evento que lo acciona es el mismo, solo se diferencia en la posición inicial del dedo, si comienza al borde de la pantalla actúa el menú desplegable, si actúa separado del borde de la pantalla salta el evento de la celda deslizante.

Cuando una celda se desliza si había otra abierta se cierra automáticamente de manera que solo puede haber una celda abierta en un instante de tiempo determinado.



**Ilustración 51: Prueba Favoritas.**



## Identificador: PF-14

Nombre:	Comprobar carrusel y enlace externo.
Descripción:	Comprobación del correcto funcionamiento de los enlaces externos y del carrusel.
Necesidad:	Baja.
Prioridad:	Baja.
Fuente:	Encargado de pruebas, Analista.

**Tabla 131: Prueba Funcional 14.**



Esta prueba se aplica en la vista de Promociones y trata de probar cómo funciona la redirección y la apertura del navegador para comprobar más datos de las promociones ofertadas.

Al pulsar la oferta se abre una ventana del navegador con la dirección de la oferta.

El carrusel tiene que ser fluido y el cambio de sección debe de funcionar correctamente.

**Ilustración 52: Prueba Promociones.**

## Identificador: PF-15

Nombre:	Comprobar ruleta y llamadas.
Descripción:	Comprobación del correcto funcionamiento de los elementos de la vista Contáctanos.
Necesidad:	Media.
Prioridad:	Media.
Fuente:	Encargado de pruebas, Analista.

**Tabla 132: Prueba Funcional 15.**



Esta prueba se aplica en la vista de Contáctanos y trata de probar como funciona las llamadas y la geo-localización de las oficinas ESES.

Al pulsar en la ruleta queda marcado un campus, si posteriormente pulsamos llamar, podemos ver como se realiza una llamada al número de la oficina que hemos marcado. Del mismo modo ocurre si pulsamos sobre el mapa, nos redirige a un mapa situado en la oficina de ESES correspondiente al campus marcado.

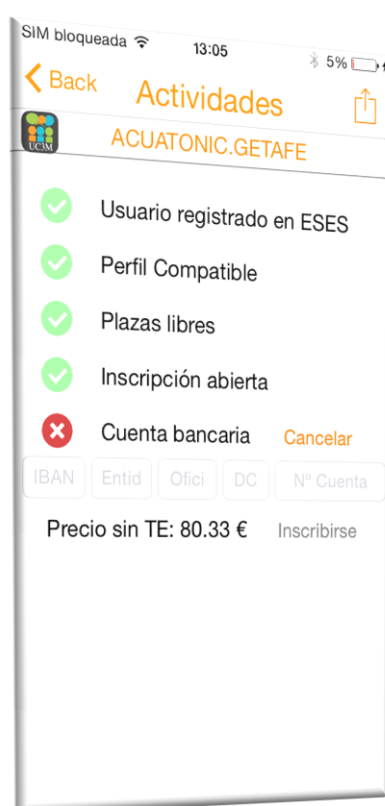
Se comprueba que no deja llamar si no existe cobertura o SIM.

**Ilustración 53: Prueba Contacto.**

## Identificador: PF-16

Nombre:	Comprobar inscripción, compra y número de cuenta.
Descripción:	Comprobación del correcto funcionamiento de las inscripciones las compras y el funcionamiento del número de cuenta.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Fuente:	Encargado de Seguridad, Encargado de pruebas, Analista.

**Tabla 133: Prueba Funcional 16.**



Esta prueba es una de las más importantes puesto que hay que comprobar muchos parámetros, las mismas comprobaciones se realizan para la vista de Comprar TE.

El primero de ellos es si corresponde el precio, con lo mostrado anteriormente en la vista actividades.

El segundo la funcionalidad de los botones cambiar/cancelar e inscribirse, tienen que estar activos cuando pueden ser útiles y desactivarse cuando pueden provocar un error.

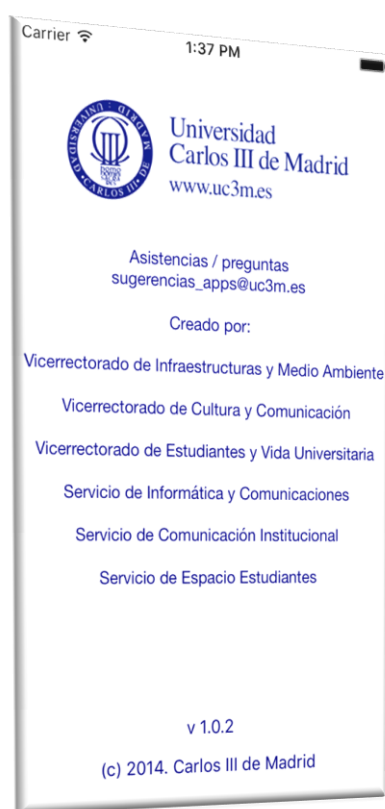
También hay que comprobar los campos de texto y sus restricciones y como afecta su relleno a la visibilidad de los botones.

**Ilustración 54: Prueba Inscripción.**

## Identificador: PF-17

Nombre:	Comprobar Información.
Descripción:	Comprobación el número de versión.
Necesidad:	Baja.
Prioridad:	Baja.
Fuente:	Encargado de pruebas, Analista.

**Tabla 134: Prueba Funcional 17.**



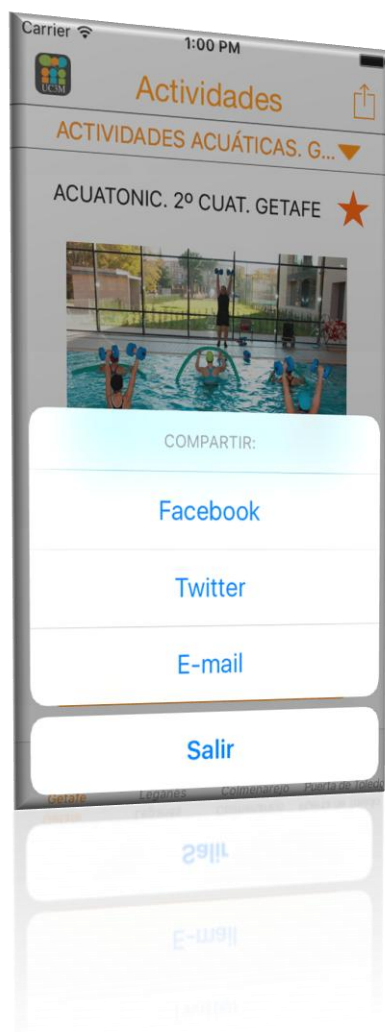
Hay que verificar que el número de versión se actualiza correctamente al cambiar en el fichero específico de Xcode en la aplicación.

**Ilustración 55: Prueba Información.**

## Identificador: PF-18

Nombre:	Comprobar Compartir.
Descripción:	Comprobación de la compartición en redes sociales en todas las vistas donde se encuentra disponible la opción y ver que se abre tanto con cliente instalado como en navegador.
Necesidad:	Baja.
Prioridad:	Baja.
Fuente:	Encargado de pruebas, Analista.

**Tabla 135: Prueba Funcional 18.**



**Ilustración 56: Prueba Compartir.**

Hay que verificar que se realiza la compartición en redes sociales comprobando que se generan los tweets y se publican en Facebook las actividades, así como comprobar que se abre correctamente el cliente y el navegador en caso de no tener cliente.

### 5.2.2 MATRIZ DE TRAZABILIDAD.

Tras analizar todas las pruebas, hemos de verificar que todos los requisitos tienen una relación directa con ellas, es decir todo requisito de sistema no funcional debe de complementarse con una prueba de los test.

	PF- 01	PF- 02	PF- 03	PF- 04	PF- 05	PF- 06	PF- 07	PF- 08	PF- 09	PF- 10	PF- 11	PF- 12	PF- 13	PF- 14	PF- 15	PF- 16	PF- 17	PF- 18
RSF – 01	X	X																
RSF – 02	X	X																
RSF – 03	X	X																
RSF – 04	X	X																
RSF – 05											X							
RSF – 06																		X
RSF – 07								X										
RSF – 08												X						
RSF – 09				X	X		X		X	X								
RSF – 10																X		
RSF – 11																X		
RSF – 12																X		
RSF – 13						X			X				X					
RSF – 14				X			X		X									

RSF – 15									X				X					
RSF – 16				X			X		X									
RSF – 17														X				
RSF – 18														X				
RSF – 19															X			
RSF – 20			X															

Tabla 136: Matriz RSF – PF.

### 5.2.3 PRUEBAS DE ESTRÉS.

A continuación se indicará cada una de las pruebas de estrés realizadas, estas pruebas se realizan a nivel de aplicación completa y muestran la velocidad de nuestra aplicación frente a escrituras y lecturas desde la base de datos, así como la velocidad de los Web-Services dependiendo de las conexiones de red.

#### Identificador: PE-01

Nombre:	Comprobar escritura con conexión Wi-Fi de 54Mbps.
Descripción:	Escritura de 10, 20, 30 y 40 inscripciones simultaneas.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Resultado:	Tiempos: <ul style="list-style-type: none"> <li>• 10 inscripciones: 0,7 segundos.</li> <li>• 20 inscripciones: 1,2 segundos.</li> <li>• 30 inscripciones: 2,1 segundos.</li> <li>• 40 inscripciones: 3,3 segundos.</li> </ul>

Tabla 137: Prueba Estrés 1.

#### Identificador: PE-02

Nombre:	Comprobar escritura con conexión Wi-Fi de 20Mbps.
Descripción:	Escritura de 10, 20, 30 y 40 inscripciones simultaneas.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Resultado:	Tiempos: <ul style="list-style-type: none"> <li>• 10 inscripciones: 1,9 segundos.</li> <li>• 20 inscripciones: 2,7 segundos.</li> <li>• 30 inscripciones: 3,5 segundos.</li> <li>• 40 inscripciones: 4,3 segundos.</li> </ul>

Tabla 138: Prueba Estrés 2.



**Identificador: PE-03**

Nombre:	Comprobar lectura con conexión Wi-Fi de 54Mbps.
Descripción:	Lectura de 50, 100, 150 y 200 Actividades simultaneas.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Resultado:	Tiempos: <ul style="list-style-type: none"><li>• 50 Actividades: 0,5 segundos.</li><li>• 100 Actividades: 0,9 segundos.</li><li>• 150 Actividades: 1,5 segundos.</li><li>• 200 Actividades: 2,3 segundos.</li></ul>

**Tabla 139: Prueba Estrés 3.****Identificador: PE-04**

Nombre:	Comprobar lectura con conexión Wi-Fi de 10Mbps.
Descripción:	Lectura de 50, 100, 150 y 200 Actividades simultaneas.
Necesidad:	Muy Alta.
Prioridad:	Muy Alta.
Resultado:	Tiempos: <ul style="list-style-type: none"><li>• 50 Actividades: 3,9 segundos.</li><li>• 100 Actividades: 7,7 segundos.</li><li>• 150 Actividades: 13,5 segundos.</li><li>• 200 Actividades: 18,3 segundos.</li></ul>

**Tabla 140: Prueba Estrés 4.**



## CAPÍTULO 6: PLANIFICACIÓN Y PRESUPUESTO

A continuación analizaremos el tipo de planificación que se seguirá para realizar cada una de las diferentes fases del desarrollo de nuestro proyecto, además debemos de estipular los costes aproximados de cada fase para saber si la aplicación es viable con el presupuesto actual y en que fases se puede invertir más y en cuales se puede recortar para mejorar nuestra aplicación. Podremos observar la planificación de una manera más intuitiva mediante un diagrama de Gantt y mediante las tablas podremos observar el desglose de todos los costes.

## 6.1 PLANIFICACIÓN

A continuación iniciaremos el proceso de planificación, cabe destacar que como en todo proyecto existen unos límites de tiempo y unas fechas de entrega que sin embargo no afectan al cumplimiento obligatorio de los derechos del trabajador por lo cual en ninguna de las fases de la planificación se incluyen los días vacacionales por días festivos nacionales ni locales, así como los 22 días de vacaciones anuales repartidas entre agosto y diciembre.

Las fases por las que concurre nuestro proyecto son 8:

- **Propuesta:** El departamento de Espacio Estudiantes propone al departamento de desarrollo de aplicaciones la realización de una aplicación móvil que mejore sus servicios y ofertas.
- **Estudio de viabilidad:** El departamento de desarrollo de aplicaciones estipula un tiempo y un coste de realización y se analiza con el departamento de economía.
- **Documento inicial:** El departamento de Espacio Estudiantes realiza un primer documento con una idea inicial de las funcionalidades que le gustaría incluir en la aplicación, así como un Storyboar de los primeros bocetos de vista.
- **Análisis:** Se unen el departamento de desarrollo de aplicaciones junto con el departamento de espacio estudiantes y se analiza el documento inicial, analizando cada una de las funcionalidades, su viabilidad, los posibles cambios y mejoras y se llega a un acuerdo sobre la mejora del documento inicial. Una vez que sabemos “el qué vamos a hacer”, pasaremos al “como debemos de hacerlo”.
- **Diseño:** de nuevo se unen los departamentos, como en todo proyecto la colaboración y el entendimiento son necesarios para conseguir un buen resultado así que las reuniones entre departamentos deben de ser constantes, se comprueba cómo se realizará la aplicación, que sistemas se utilizarán como deben de ser las conexiones entre diferentes entornos así como el flujo de los datos.

- Implementación: es una de las fases más delicadas, en esta fase se hacen realidad las ideas anteriormente descritas, es por ello que cualquier error en las anteriores fases se ve afectado de forma exponencial en esta y es imprescindible tener las anteriores sin fisuras ni dudas.
- Pruebas: Como en todas las fases, en la implementación también se producen errores, errores que no pueden formar parte del producto final ya que es el cliente quien percibirá dichos errores, por ello se realizan pruebas de rendimiento, de funcionalidad y de estrés para estar seguros de que el producto que llega a las manos del consumidor es de calidad y hace lo que debe de hacer.
- Documentación: es la última fase de un proyecto y una de las más infravaloradas pero cuando un proyecto pasa de mano en mano por diferentes departamentos y diferentes personas, la documentación es la pieza clave del entendimiento y una guía básica de la aplicación, sin ella cualquier versión nueva, cualquier cambio o cualquier duda sería muy complicada de resolver o realizar.

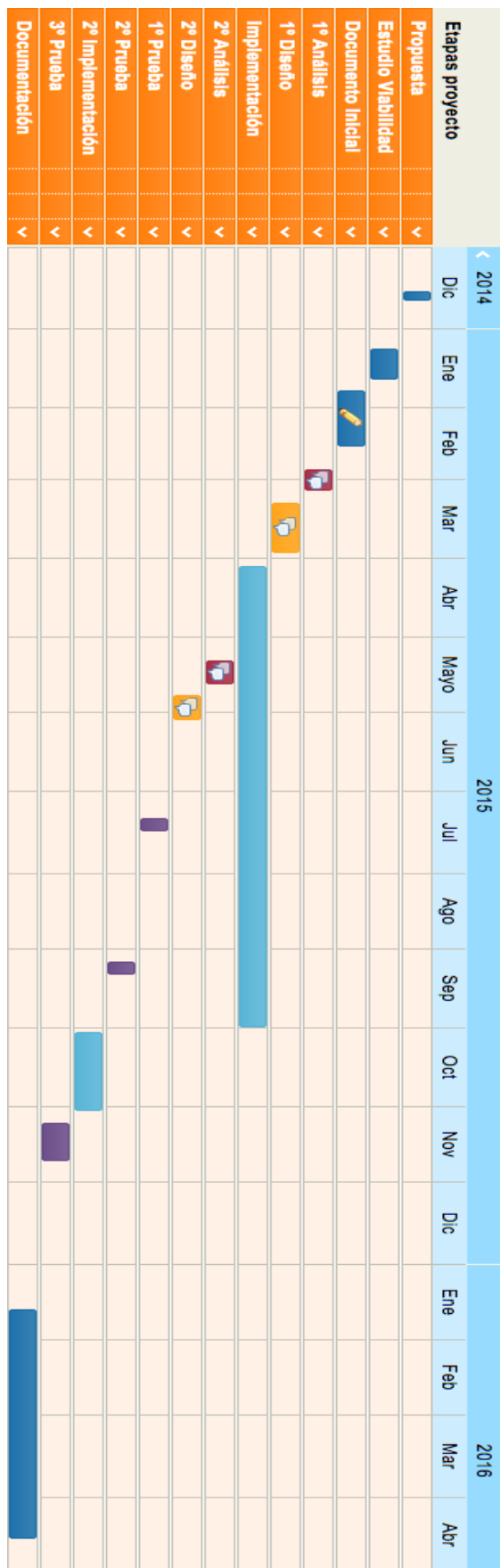
A continuación se muestra una tabla con la duración de cada una de las fases, así como el día en que se comenzó y se finalizó cada fase. Se dice que el proyecto perfecto debería de cerrar cada fase antes de empezar la siguiente, pero como se dijo anteriormente, a menudo se comenten errores o pequeños cambios que no se habían contemplado y que deben de ser modificados, además afectan a las posteriores fases y como en todo proyecto real, en este también ha sucedido.

Sin embargo la fase de implementación no se ha visto afectada debido a que los cambios realizados no afectaban a la implementación hasta la fecha realizada.

Actividad	Fecha Inicio	Fecha Fin	Duración en días
Propuesta	17/12/14	19/12/14	2
Estudio de la viabilidad	08/01/15	20/01/15	9
Documentación inicial	25/02/15	16/02/15	17
1º Análisis	25/02/15	04/03/15	7
1º Diseño	10/03/15	29/03/15	11
Implementación	04/04/15	30/09/15	116
2º Análisis	11/05/15	19/05/15	5
2º Diseño	24/05/15	02/06/15	8
1º Pruebas	11/07/15	13/07/15	2
2º Pruebas	05/09/15	08/09/15	3
Implementación Corrección de errores	3/10/15	02/11/15	21
3º Pruebas	07/11/15	21/11/15	10
Documentación	20/01/16	18/04/16	61

Tabla 141: Fases del proyecto y duración.

### 6.1.1 DIAGRAMA DE GANTT



## 6.2 PRESUPUESTO

El siguiente punto detalla uno de las características más importantes a la hora de realizar un proyecto y es que todo proyecto cuesta unos recursos y dependiendo de ellos podemos ampliar o rebajar nuestras expectativas, en nuestro caso el presupuesto no es un hándicap demasiado importante puesto que la universidad cuenta con una gran capacidad de recursos y tiene a los departamentos trabajando de manera continua, de manera que es más importante las prioridades de cada proyecto que los costes de los mismo puesto que tarde o temprano se acabarán realizando si se trata de necesidades o mejoras.

De todas las maneras, la universidad lleva un registro de gastos y también tiene que cumplir con los presupuestos estipulados, así que a continuación se mostrará el desglose en horas trabajadas por fase del proyecto.

### 6.2.1 TIEMPO DE FASES

Gracias a la tabla de días trabajados por fase del proyecto podemos calcular de manera instantánea los costes en horas que se han realizado.

Actividad	Duración en días	Horas trabajadas por día	Horas totales trabajadas
Propuesta	2	8	16
Estudio de la viabilidad	9	8	72
Documentación inicial	17	8	136
1º Análisis	7	8	56
1º Diseño	11	8	88
Implementación	116	8	928
2º Análisis	5	8	40



2º Diseño	8	8	64
1º Pruebas	2	6	12
2º Pruebas	3	6	18
Implementación Corrección de errores	21	8	168
3º Pruebas	10	6	60
Documentación	61	8	488
Total	272	-	2146

Tabla 142: Horas trabajadas por fase de proyecto.

### 6.2.2 COSTE DEL PERSONAL.

Los gastos de un proyecto se dividen en 3 partes, la primera los costes de personal, la segunda los costes de hardware y la tercera los costes de software, en este apartado nos encargaremos de la primera de ellas que queda representada en la siguiente tabla:

Actividad	Categoría	Dedicación (Hombre/Mes) en €	Coste (Hombre/Mes) en €	Coste en €
Sánchez Muñoz, Daniel	Analista	143,33	2100	11.838,5
Sánchez Muñoz, Daniel	Diseñador	143,33	1800	1.908,8
Sánchez Muñoz, Daniel	Programador	143,33	1200	9.176,2
Sánchez Muñoz, Daniel	Tester	107,5	800	669,8

Tabla 143: Coste personal.

Si un año tiene 365 días y quitamos los 119 que hay entre festivos nacionales, festivos locales y sábados y domingos nos quedan 237 días laborables, de estos días un trabajador puede disfrutar de 22 días de vacaciones lo cual nos da un resultado de 215 días productivos por una persona en un año, si esos días los multiplicamos por 8 horas diarias que es lo que tiene una jornada laboral completa, nos da un total de 1.720 horas producidas por un hombre en un año, ahora dividimos el número de horas de producción entre los 12 meses de un año y tenemos un total de 143,33 horas al mes de producción por un hombre.

$$\text{Coste} = \frac{\text{Nº Horas trabajadas en el proyecto}}{\text{Dedicación hombre/mes}} \times \text{Salario mensual}$$

El coste de cada uno de los puestos se calcula de la siguiente manera, el trabajo del Analista es de un total 101 horas por 8 horas diarias es igual a 808 horas, estas horas divididas entre la dedicación hombre/mes de 143,33 horas es igual a 5,63, factor que multiplicaremos por el salario mensual del analista 2.100€ dando un total de 11.838,50€.

$$\text{Coste Analista} = \frac{808 \text{ horas}}{143,33 \text{ horas}} \times 2100\text{€} = 11.838,50\text{€}$$

$$\text{Coste Diseñador} = \frac{152 \text{ horas}}{143,33 \text{ horas}} \times 1800\text{€} = 1.908,80\text{€}$$

$$\text{Coste Programador} = \frac{1096 \text{ horas}}{143,33 \text{ horas}} \times 1200\text{€} = 9.176,20\text{€}$$

$$\text{Coste Tester} = \frac{90 \text{ horas}}{107,5 \text{ horas}} \times 800\text{€} = 669,80\text{€}$$

$$\text{Coste total} = 11.838,50\text{€} + 1.908,80\text{€} + 9.176,20\text{€} + 669,80\text{€} = 23.593,30\text{€}.$$

### 6.2.3 COSTE DEL HARDWARE.

Para la realización del proyecto han sido necesarios adquirir una serie de dispositivos y hardware con el que realizar la implementación y las pruebas, como se describe en la siguiente tabla:

Dispositivo	Coste en €	% usado en el proyecto	Uso real en meses	Devaluación en meses	Coste Imputable en €
Mac Mini	1099	100	13	72	198,43
iPhone 6	659	65	8	48	71,37
iPhone 5	599	65	8	24	108,11

**Tabla 144: Coste en Hardware.**

Para calcular el coste imputable de un producto es necesario hacer una serie de cálculos, no todos los dispositivos se utilizan el mismo tiempo, se devalúan de la misma manera ni se amortizan igual, por ello no basta con sumar el coste del producto en bruto por suerte para nosotros, los productos de Apple suelen ser de una gran calidad y con el tiempo no se devalúan de la misma manera que otros dispositivos con lo que la amortización es mejor que con otros productos similares del mercado.

$$\text{Coste imputable} = \frac{\text{Uso del dispositivo}}{\text{Devaluación}} \times \text{Coste del equipo} \times \text{Uso dedicado}$$

Por lo tanto los diferentes costes imputables, como aparecen en la tabla son los siguientes.

$$\text{Coste Mac Mini} = \frac{13 \text{ meses}}{72 \text{ meses}} \times 1.099\text{€} \times 100\% = 198,43\text{€}$$

$$\text{Coste iPhone 6} = \frac{8 \text{ meses}}{48 \text{ meses}} \times 659\text{€} \times 65\% = 71,37\text{€}$$

$$\text{Coste iPhone 5} = \frac{8 \text{ meses}}{24 \text{ meses}} \times 499\text{€} \times 65\% = 108,11\text{€}$$

Como se puede apreciar, el coste total del hardware adquirido es de 377,91€.

#### 6.2.4 COSTE DEL SOFTWARE Y LICENCIAS.

Los costes de licencia son relativamente pequeños para nuestro proyecto debido a que se está utilizando mayoritariamente licencias de código abierto OpenSource y con Copyleft con lo cual no hay que pagar ninguna licencia por trabajar con ellos, como pueden ser Xcode que aunque no es de código abierto, al adquirir nuestro Mac viene de manera gratuita, se puede decir que viene incluido en el precio del dispositivo. Para la base de datos se utilizó MySQL también de código abierto y para los WebServices se utilizó el IDE de Java IDEA, para realizar algunos iconos y botones se utilizó GIMP y todos los pods de Cocoa son de código abierto.

Solo hay un coste de licencia que debemos de pagar por año, esta licencia es la de desarrollador de Apple, una persona puede ser desarrollador sin necesidad de pagar esta licencia, pero si queremos poder asociar a nuestro proyecto un dispositivo de prueba donde probar nuestra app debemos de contratar dicha licencia, además sin ella no podríamos lanzar nuestra app al market de Apple ni autorizar a los dispositivos de testeo.

El precio de dicha licencia para una persona individual es de 99€ al año, y como nuestro proyecto ha durado cerca de 15 meses, pero la parte de desarrollo y testeo no llega al año, nos sobra con contratarla una única vez sin renovar.

#### 6.2.5 COSTE TOTAL DEL PROYECTO.

Tras analizar los principales costes del proyecto, hay que incluir un margen de costes ocultos que existen y no se presupuestan directamente como parte del proyecto pero que si se cobran, estos costes extra pueden ser de diferentes índoles y cuantías, desde material administrativo hasta tasas de transportes, etc y suelen alcanzar un 10% del precio total del proyecto con lo cual el desglose de los costes del proyecto quedaría reflejado como indica la siguiente tabla.

Tipos de coste	Coste en €
Personal	23.593,30
Hardware	377,91
Software y licencias	99
Indirectos	2.407
I.V.A 21%	5.560,21
Total sin I.V.A	26.477,231
Total	32.037,44

**Tabla 145: Coste total.**

El coste total del proyecto asciende a la cifra de 32.037,44€.



## CAPÍTULO 7: CONCLUSIÓN Y PRÓXIMAS NOVEDADES.

En el capítulo número siete se tratan las conclusiones a las que se ha llegado tras concluir el proyecto, experiencias, detalles, curiosidades, expectativas y lo más importante las novedades que se desean implantar en las próximas versiones de la aplicación.

## 7.1 CONCLUSIONES.

La parte de las conclusiones son un análisis que se realiza después de concluir el proyecto para concretar cuáles han sido los puntos fuertes, los puntos débiles, que objetivos se han logrado y cuáles han resultado más dificultosos, para en posteriores proyectos poder contar con una experiencia previa de mejora o de guía de uso.

Para ello haremos dos conclusiones, unas conclusiones objetivas sobre el resultado de la aplicación y su desarrollo, y unas conclusiones personales sobre las fases y su desarrollo siempre subjetivas.

### 7.1.1 PROYECTO.

La conclusión que obtenemos al finalizar el proyecto es de una valoración muy positiva con respecto a los resultados iniciales esperados, durante todo el desarrollo del proyecto se han ido cumpliendo con las expectativas en cada paso que se daba, se realizaron vista por vista el desarrollo y las pruebas.

La dos principales características conseguidas en esta aplicación han sido, conseguir una interfaz adaptada a los estándares estéticos de creación de aplicaciones móviles de Apple, rompiendo por primera vez con los aplicados anteriormente y generando una interfaz mucho más minimalista y limpia.

Además con la ayuda de los Pods hemos conseguido funcionalidades complejas de una manera rápida y sencilla y con un control de versiones que se actualiza con cada nueva versión de iOS, este detalle nos ha dado el tiempo necesario para dedicarnos a otros procesos de la aplicación, además estamos orgullosos de haber conseguido una interfaz sencilla de utilizar sin apenas puntos de confusión para el usuario final.

La segunda característica conseguida es el control de intercambio de datos mediante los Web Services y su correcto funcionamiento, es una de las partes más fundamentales y gracias a la extensión para Chrome Postman hemos conseguido unos servicios eficientes y rápidos, sin errores en las transacciones y probados antes de proporcionarlos a la implementación del cliente.

Como se puede apreciar, hemos cuidado tanto el FrontEnd como el BackEnd, consiguiendo una armonía muy buena.

Además la planificación cuidadosa y de paso a paso ha servido para poder trabajar en paralelo con varias fases del proyecto y que mientras algunos requisitos aun no estaban cerrados, ya se estuvieran implementando



funcionalidades e incluso realizando pruebas, esto nos ha hecho ganar tiempo sin tener que esperar a que esté terminada la fase del análisis y el diseño para empezar con la implementación trabajando como una cadena de producción, claro está que cada paso de implementación que se realizaba, estaba cerrada en sus fases previas de análisis y diseño.

También cabe destacar que gracias a los Frameworks de Crashlytics y Flurry podemos asegurar que los errores reportados son mínimos y muy sencillos de reparar sin implicar consecuencias graves en el proyecto y que los informes enviados por Flurry son positivos destacando una alta afluencia de descargas en los usuarios con una alta tasa de instalaciones duraderas lo cual certifica la aceptación de los usuarios.

### 7.1.2 PERSONALES.

Cuando comencé a realizar este proyecto no sabía que un tiempo atrás se convertiría en mi trabajo fin de grado, me ha encantado la experiencia de poder tocar las dos vertientes de un proyecto, esa en la que todo se hace como en la práctica y esa en la que todo se hace como en la teoría, pues así ha sido mi experiencia, comencé realizando literalmente la práctica con sus problemas y sus búsquedas de soluciones, con su implementación y su diseño y en general manchándome las manos y viendo como poco a poco surgía de la nada una aplicación, y no una aplicación cualquiera, una aplicación que hacía cosas, cosas que no solamente iba a utilizar yo para presentar un proyecto, cosas que los alumnos y el personal de la universidad de la UC3M pueden utilizar todos los días y cosas que mejoran la vida de muchas personas y eso te hace sentir genial.

Continué con la teoría, es parte tediosa y aburrida que nadie quiere realizar pero que sin ella nada de esto sería posible, cuando realizas esta parte te das cuenta de lo importante que es la organización, la comunicación y las normas, no es atractivo pero mejora tus capacidades y poco a poco empiezas a apreciar esa parte que nadie conoce, ese trabajo que solo tú puedes ver pero que multiplica de forma exponencial tu resultado final.

Estoy orgulloso de poder ser ingeniero y realizar estas obras de arte que mejoran la vida de la gente y que te hacen sentir lleno, además agradezco poder haber realizado este reto que me ha dado una visión mucho más profunda de lo que es un proyecto en la vida real, con sus tiempos de entrega, sus problemas de última hora, sus cambios inesperados que tiran por tierra el trabajo de toda

una semana, todas estas experiencias me han hecho crecer como desarrollador pero mucho más como persona, te das cuenta de cómo es el mundo real y aprendes a resolver los problemas por ti mismo sin depender de nadie.

Concluyo diciendo que lo que sentí el día que vi todos los carteles con la foto de mi aplicación distribuidos por la universidad me lleno de orgullo y satisfacción, a la vez que de miedo por lo que los usuarios pudiesen pensar de ella.

## 7.2 PRÓXIMAS NOVEDADES.

Cuando un proyecto ha finalizado siempre nos queda la sensación de haber podido mejorar en algo nuestra aplicación, todo funciona correctamente y hace lo que debe de hacer pero sentimos que necesitamos más, a menudo sucede que esos extras se intentan añadir durante el proceso de creación de la aplicación, alargando el proceso y haciendo más costosa su puesta en explotación.

Sin embargo todas esas nuevas funcionalidades o cualidades son mejoras que no debemos de desechar y que pueden incluirse en las futuras versiones de la aplicación, por eso a continuación haremos un listado de las nuevas mejoras que se desean aplicar a Campus Life.

- Hacer reservas, actualmente la aplicación solo permite inscribirse en actividades deportivas y culturales pero no permite hacer reservas de canchas, pistas o campos de juego.
- Recibir notificaciones, se desea que la aplicación reciba notificaciones personalizadas según las actividades inscritas del usuario y sus gustos personales como puedan ser ofertas de interés.
- Añadir búsqueda de recorrido en los mapas, se desea añadir la funcionalidad de encontrar la ruta deseada hasta la oficina marcada en el mapa.
- Añadir una sección con información sobre los centros deportivos, instalaciones, etc. Para mejorar la sensación de conocer mejor el centro deportivo intentamos que el cliente pueda ver de antemano las instalaciones y sus servicios.

- Mejora en RRSS, se pretende añadir más redes sociales y mejorar el servicio de compartición en las redes, para realizar una mejor interacción entre aplicaciones de esta índole.
- Descarga de recibos y resguardos de inscripción y compra de tarjetas.
- Vista en tiempo real de las listas de espera, poder ver en tiempo real las solicitudes de una actividad y saber cuántas plazas quedan libres y cuántas necesitan ser descartadas para poder inscribirnos en dicha actividad.

Estas son algunas de las mejoras que se desean aplicar en versiones futuras de la aplicación, algunas se materializaran en la próxima versión, algunas se materializaran en posteriores versiones, algunas de las actuales podrían desaparecer y otras jamás llegaran a implementarse, todo depende del progreso de la aplicación, su aceptación, uso y necesidades, lo que está claro es que habrán nuevas versiones y que se mejorará para que el cliente este más satisfecho cada día.



## GLOSARIO.

- Android: es un sistema operativo móvil de la multinacional Google. Originalmente desarrollado para dispositivos fotográficos, después se ha usado en dispositivos smartphone.
- App: término inglés y abreviado para referirse a aplicaciones para teléfonos móviles inteligentes.
- Apple: es una empresa multinacional estadounidense que diseña y produce equipos electrónicos y software.
- BackEnd: Parte del desarrollo del servidor.
- Campus Life: Nombre de la aplicación expuesta en el presente documento.
- Carousel: Función deslizable que permite ver varias vistas ocultas girando haciendo el efecto de un carrusel que se mueve.
- Chrome: Navegador de Google.
- Copyleft: software que no posee derechos de autor.
- Crashlytics: Software para el reporte de errores.
- Drawer: Función deslizable que permite ver una vista oculta en un lateral de la pantalla.
- ESES: Acrónimo del departamento y servicio Espacio Estudiantes.
- Facebook: Red social.
- Flurry: Software para el control de estadísticas de uso.
- Framework: Marco de trabajo, código externo que nos ayuda aplicando unas funcionalidades extra predefinidas.

- FrontEnd: Parte del desarrollo del cliente.
- Google: es una empresa multinacional estadounidense que diseña y produce equipos electrónicos y software.
- HUD: elemento de programación para denotar una carga de datos.
- http: protocolo de transferencia de hipertexto utilizado para las comunicaciones vía web.
- IDE: entorno de desarrollo integrado, nos ayuda en la creación de un proyecto y en el desarrollo de las aplicaciones mejorando la experiencia.
- iOS: es un sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), después se ha usado en dispositivos como el iPod touch y el iPad.
- Java: Lenguaje de programación orientado a objetos.
- JSON: Formato de texto ligero para el intercambio de datos.
- Label: etiqueta de texto en una interfaz.
- Login: Credenciales, autenticación en un sistema.
- Loombok: Software para la gestión de anotaciones.
- Market: mercado de aplicaciones móviles.
- Microsoft: es una empresa multinacional estadounidense que diseña y produce equipos electrónicos y software.
- MySQL: Sistema gestor de base de datos con licencia GPL.
- OpenSource: Software de código abierto.

- Parsear: Convertir estructuras de datos de un formato a otro.
- Play: Elemento de reproducción multimedia.
- Pop-Up: Ventana emergente.
- Postman: Software para trabajar con web-services.
- Pulsable: que se puede pulsar.
- RRSS: Redes Sociales.
- Scroll: Acción de deslizar una vista hacia arriba o hacia abajo para poder ver todo el contenido de la misma.
- SGBD: Sistema gestor de base de datos.
- Smartphone: Dispositivo inteligente de última generación conectado a la red.
- Storyboard: Diagrama de interfaces de la aplicación.
- UC3M: Acrónimo de la Universidad Carlos III de Madrid.
- Tab: Pestaña de una vista que permite cambiar el contenido de la misma.
- TE: Acrónimo de la Tarjeta Espacio Estudiantes.
- Tester: Persona dedicada a la sección de pruebas.
- TextField: Campo de texto en una interfaz.
- Time Out: Denota que la carga de datos está fuera de tiempo y se paraliza, necesitando una recarga.
- Token: Código alfanumérico que identifica a un usuario en relación a una aplicación concreta o a un servicio concreto.



- Twitter: Red social.
- Web Service: Servicio web, que es capaz de comunicar dos aplicaciones entre sí realizando un intercambio de datos mediante el protocolo http.
- Wi-Fi: Tecnología de conexión inalámbrica.





## BIBLIOGRAFÍA.

- [1] Cigarran Recuero, Juan Manuel. Aprende iOS: Primeros Pasos. Juan Manuel Cigarran Recuero 2012.
- [2] Cigarran Recuero, Juan Manuel . Aprende iOS: Un Poco de Objective – C. Juan Manuel Cigarran Recuero 2013
- [3] Cigarran Recuero, Juan Manuel . Aprende iOS: Más Objective – C. Juan Manuel Cigarran Recuero 2014
- [4] Cigarran Recuero, Juan Manuel . Aprende iOS: Foundation Framework. Juan Manuel Cigarran Recuero 2015
- [5] Apple Inc. Guía de referencia sobre la implantación de iOS. Apple 2016.
- [6] Smith, Nick. Xcode Primer: Starting Objective – C. AppSmith Books DK 2012.
- [7] Hawking, Jeff. Cocoa and Objective – C: CookBook. Packt Publishing 2013.
- [8] Purushothaman, Jobinesh. RESTful Java Web Services. Packt Publishing 2015.
- [9] Rene, Enriquez. RESTful Java Web Services Security. Packt Publishing 2014.
- [10] Dewailly, Ludovic. Building a RESTful Web Service with Spring. Packt Publishing 2015.
- [11] Portal de deporte y cultura de la UC3M.  
[http://portal.uc3m.es/portal/page/portal/cultura\\_y\\_deporte/eses/campus\\_life](http://portal.uc3m.es/portal/page/portal/cultura_y_deporte/eses/campus_life)
- [12] Descarga de la aplicación móvil Campus Life para iOS.  
<https://itunes.apple.com/es/app/campus-life-uc3m/id904190258?mt=8>
- [13] Descarga de la aplicación móvil Campus Life para Android  
<https://play.google.com/store/apps/details?id=es.uc3m.espacioestudiantes&hl=es>
- [14] Artículo wiki sobre la empresa Apple Inc.  
<https://es.wikipedia.org/wiki/Apple>
- [15] Artículo wiki sobre el término App.  
<https://es.wikipedia.org/wiki/APP>
- [17] Artículo wiki sobre el término sistema operativo móvil.  
[https://es.wikipedia.org/wiki/Sistema\\_operativo\\_m%C3%B3vil](https://es.wikipedia.org/wiki/Sistema_operativo_m%C3%B3vil)
- [16] Artículo wiki sobre el sistema operativo Android.  
<https://es.wikipedia.org/wiki/Android>
- [16] Artículo wiki sobre el sistema operativo iOS.  
<https://es.wikipedia.org/wiki/IOS>
- [16] Artículo wiki sobre el sistema operativo Windows Phone.  
[https://es.wikipedia.org/wiki/Windows\\_Phone](https://es.wikipedia.org/wiki/Windows_Phone)
- [16] Artículo wiki sobre el sistema operativo Black Berry.  
[https://es.wikipedia.org/wiki/BlackBerry\\_OS](https://es.wikipedia.org/wiki/BlackBerry_OS)
- [16] Artículo wiki sobre el sistema operativo Symbian.  
<https://es.wikipedia.org/wiki/Symbian>
- [16] Artículo wiki sobre el sistema operativo Firefox OS.  
[https://es.wikipedia.org/wiki/Firefox\\_OS](https://es.wikipedia.org/wiki/Firefox_OS)

- [16] Artículo wiki sobre el sistema operativo Ubuntu Touch.  
[https://es.wikipedia.org/wiki/Ubuntu\\_Touch](https://es.wikipedia.org/wiki/Ubuntu_Touch)
- [16] Artículo wiki sobre el IDE Xcode.  
<https://es.wikipedia.org/wiki/Xcode>
- [16] Artículo wiki sobre el lenguaje de programación Objective-C.  
<https://es.wikipedia.org/wiki/Objective-C>
- [16] Artículo wiki sobre el gestor de dependencias CocoaPods.  
<https://en.wikipedia.org/wiki/CocoaPods>
- [16] Artículo wiki sobre el servidor de aplicaciones Tomcat.  
<https://es.wikipedia.org/wiki/Tomcat>
- [16] Artículo wiki sobre el gestor de base de datos MySQL.  
<https://es.wikipedia.org/wiki/MySQL>
- [18] Artículo sobre los sistemas operativos móviles más usados.  
<http://www.expansion.com/economia-digital/companias/2015/12/09/56684be1ca474151018b4590.html>
- [19] Usuarios con Android en el mundo.  
<http://www.cnet.com/es/noticias/google-1400-millones-usuarios-android/>
- [20] Gestión de dependencias.  
<http://www.miguel diazrubio.com/desarrollo-ios-gestionar-dependencias-con-cocoapods/>
- [21] Artículo sobre REST.  
<http://www.arquitecturajava.com/servicios-rest/>
- [22] Artículo sobre MVC.  
<http://www.desarrolloweb.com/articulos/que-es-mvc.html>
- [23] Artículo sobre Cliente-Servidor.  
<https://www.ecured.cu/Cliente-Servidor>
- [24] Artículo sobre Instalación de Tomcat.  
[http://chuwiki.chuidiang.org/index.php?title=Instalaci%C3%B3n de Tomcat en Windows](http://chuwiki.chuidiang.org/index.php?title=Instalaci%C3%B3n_de_Tomcat_en_Windows)
- [25] Artículo sobre Instalación deMySQL.  
<http://tecnologia.uncomo.com/articulo/como-instalar-mysql-en-windows-23822.html>
- [26] Artículo sobre Instalación de SpringBoot.  
<https://unpocodejava.wordpress.com/2013/09/05/un-poco-de-spring-boot/>



## ABSTRACT.

This project describes the development of Campus Life, an application that supports Apple's mobile platform (iOS ), designed to provide solutions to the needs of the department Students Area (ESES) belonging to the Carlos III of Madrid (UC3M). It provides inquiry and registration on sports, cultural and orientation activities organized by ESES.

Campus Life is a free application and educational environment but is not accessible to all users, the application is restricted to students, teachers and administrative staff and services UC3M access.

Its main features and services are:

- Apply for a card (TE), giving discounts University and benefits of ISIC/ITIC card.
- Consult promotions.
- Inform the offer of activities , which are organized by campus and types of activity . Each activity can be found a picture a brief description , schedule and price.
- Enroll in the activities.
- Mark activities as favorites to register more later or share on social networks.
- Access your registered activities.
- Contact ESES department and geotag their locations in different campus and access their profiles on different social networks.

The current document details the complete process needed to create the project.

# INTRODUCTION AND OBJECTIVES.

We will discuss the motivations and objectives that have pushed us to make the Campus Life project, analyzing its development step by step from analysis to testing, through design and implementation. To make it easier comprehension a brief summary at the beginning of each section will give an overview of the topic will be held.

## 1.1 OVERVIEW.

The primary need that should cover the Campus Life application is managing much of the procedures that can be performed on the website of the Department of ESES (Space Students) transferred to the environment iOS mobile devices. You can query and registration on sports, cultural and orientation activities.

The main task that will perform the application is allowed to enroll in activities that offer ESES and to display the offer by campus university staff, also can be purchased quickly and easily different cards, in addition to different services will be included management and consulting promotions, contact and favorite activities.

## 1.2 MOTIVATION.

In today's world, is experiencing a period of overwhelming development in the field of new technologies and communications, in this significant change is necessary to update all the services we have and the new ones are generated quickly and continuously, not be up to date on new services, benefits or trends, you may trigger a massive abandonment by customers using our products and result in a failure for our projects.

The current rise in sales of smartphones or smart mobile devices, indicates that today the use of mobile phone is much more dynamic and offers a lot of

possibilities approaching closely to the services it can offer a laptop or desktop and in some cases surpassing power and performance to the latter.

Therefore the mobile application development is being the new trend in the field of development, migrate all those services previously users were offered via browser or desktop client and now must be optimized to the needs offering new phones intelligent.

So the main motivation for this project is to upgrade the services offered ESES via the web to an environment booming as mobile applications, providing the user making representations from any platform, this also allows us to reach a more users, actualize and keep up on the latest technology trends.

### 1.3 OBJETIVES.

The main objective of the Campus Life application is attempting to move all the possibilities offered by the web portal Student Space trying to improve the usability of users adding native functionality of our smart mobile devices, and is intended to facilitate perform the relevant tasks from anywhere at any time thanks to the connectivity offered by these devices.

As immediate objectives is to:

- Create a native application for iOS browser eliminating dependence and taking advantage of specific features of smartphones.
- To design a quick and easy application to use for all users.
- Create an application as light as possible and not consume too many resources, so that any type of mobile device characteristics are slim that can use it.
- It should allow shopping, easily and safely.
- Display whole range of activities of all campuses.

- It must allow make inscriptions activities.
- Will offer promotions and discounts.
- Controlling user errors to minimize failures caused by improper use and prevent the user frustrations.

#### 1.4 DEVELOPMENT STAGES.

The project is divided into six stages of development.

The first phase involves studying technologies which will be similar products and development platforms to test the feasibility of the project., After a thorough analysis platform to use, its architecture and its characteristics will be decided.

After selecting the platform we enter phase two, with the analysis of the requirements to be met by the system for this purpose several meetings with the Department of Student Space will be made and needs are analyzed, the feasibility of the requirements and priority the same.

When we complete the list of requirements, we will enter phase three, dedicated to the design of architecture, indicating that component covers every need of the requirements.

The fourth phase of development is the implementation of application code, it is essential that the previous phases are closed and well designed as a change in any of them drag a mistake that would be very difficult to correct, delaying the delivery of the project and firing their costs.

The fifth phase is the testing phase and may overlap with the implementation phase, it is advisable to perform tests according to specific implementations of each function performed since leaving tests by the end of the implementation can lead to very complex code corrections, still there is evidence that full implementation need to be made.



The last phase is the preparation of this document explaining in detail the entire process of creating the application and that is necessary for the release of new versions and study by new components in its creation.

## **1.5 DOCUMENT STRUCTURE.**

Then the purpose of each of the chapters in which the document is divided briefly outlined.

### **1.5.1 CHAPTER 1, Introduction and objectives.**

Its purpose is to give an overview of the project, which have been the objectives and motivations necessary to undertake the project, so you better understand each of the following chapters.

### **1.5.2 CHAPTER 2, State of affairs.**

Its purpose is to analyze the different existing technologies and platforms reaching a consensus which is the option that best fits the needs of our project and choosing on which it will develop.

### **1.5.3 CHAPTER 3, Analysis.**

Its purpose is to conduct a thorough process in requirements analysis, requirements that our application must meet to satisfy all the requirements agreed with space Students in the preparation thereof, and use cases that may occur in the application.

### **1.5.4 CHAPTER 4, Design.**

Its purpose is to define the application design adapted to the agreed requirements, architecture and components used, your database and storage structures and customer interface that will be what ultimately the end user can see.

#### **1.5.5 CHAPTER 5, Implementation and testing.**

Its purpose is to analyze the implementation, is not to show the application code but rather the type of design that has been used when programming and logical structure and utilities that have helped us in the process. testing will also be conducted to verify proper operation of the application.

#### **1.5.6 CHAPTER 6, Planning and budgeting.**

Its purpose is to study the process and project planning by calculating what have been the costs incurred in the purchase of equipment, licenses and salaries of workers, as well as costs or additional expenses, you must calculate the hours devoted to each phase of project and make a diagram showing the production line project.

#### **1.5.7 CHAPTER 7, Conclusions and coming soon.**

Its purpose is to reach a subjective and objective conclusion of the work done during the different phases of the project as well as future functionalities that could be developed in new versions that would complement or enhance the services provided.

# CONCLUSION AND COMING SOON.

In chapter number twelve the conclusions that have been reached after completion of the project, experiences, details, curiosities, expectations and most importantly the developments that wish to implement in the next versions of the application are discussed.

## 2.1 CONCLUSIONS.

Part of the findings are an analysis that is performed after completion of the project to realize what were the strengths, weaknesses, which objectives have been achieved and which have been more difficult for subsequent projects to have an experience prior improvement or user guide.

To do this we will make two conclusions, some objective conclusions about the outcome of the application and development, and a personal conclusions and development phases always subjective.

### 2.1.1 PROJECT.

The conclusion we get at the end of the project is a very positive assessment regarding the initial expected results throughout the project have been meeting expectations in every step you took, they were held view development and tests.

The two main features achieved in this application have been adapted to achieve the aesthetic standards of creating mobile applications Apple, breaking for the first time with the previously applied and generating a much more minimalist and clean interface.

And with the help of Pods we have managed complex functionality quickly and easily and with version control that is updated with each new version of iOS, this detail has given us the necessary time to devote to other application processes, we are also proud to have achieved a simple interface to use with little points of confusion for the end user.

The second feature is the control of data exchange through the Web Services and its proper functioning is one of the most fundamental parts and thanks to the Chrome extension Postman have achieved some efficient and fast services

without errors in transactions and tested before to provide them to the client implementation.

As you can see, we care both frontend and the backend, getting a very good harmony.

Besides the careful and step planning step has served to work in parallel with various phases of the project and that while some requirements were not yet closed, and were being implemented functionalities and even carrying out tests, this has made us gain time without wait for this phase completed analysis and design to implementation to start working as a production line, it is clear that each implementation step that was made, was closed at its previous stages of analysis and design.

It is also noteworthy that thanks to the Frameworks of Crashlytics and Flurry we can ensure that the errors reported are minimal and very simple to repair without involving you record consequences on the project and the reports sent by Flurry are positive highlighting a high influx of downloads on users with a high rate of durable installations which certifies the user acceptance.

### 2.1.2 PERSONAL.

When I started this project did not know that a while back would become my job to grade, I loved the experience of touching the two sides of a project, the one where everything is done and in practice and that in that everything is done as in theory, it has thus been my experience, I began literally making practice with their problems and search for solutions to their implementation and design and overall staining my hands and seeing how slowly emerged of nowhere an application, and not any particular application, an application that did things, things that are not only going to use me to present a project, things that students and university staff UC3M can use every day and things that improve the lives of many people and it makes you feel great.

I continued with the theory, is tedious and boring part that nobody wants to do but without her none of this would be possible, when you do this part you realize how important it is organization, communication and standards, is not attractive but improves your skills and slowly begin to appreciate that part that nobody knows, that work that only you can see but that multiplies exponentially your final result.

I am proud to be an engineer and perform these works of art that enhance the lives of people and make you feel full, also thank to have made this challenge has given me a much deeper insight into what a project in the real life, with delivery times, problems of last minute, their unexpected changes pulling the ground work for a week, all these experiences have made me grow as a developer but much more as a person, you realize how the real world and learn to solve problems by yourself without depending on anyone.

I conclude by saying that what I felt the day I saw all the posters with the picture of my application distributed by the university I am filled with pride and satisfaction, while fear so users could think about it.

## 2.2 COMING SOON.

When a project has always finished we have the feeling of being able to improve on something our application, everything works fine and does what it should do but we feel we need more, it often happens that these extras are trying to add during the creation process application, lengthening the process and making it more expensive to put into operation.

However, all these new features or qualities are improvements that we should not discard and which may be included in future versions of the application, so then we will make a list of the improvements that you want to apply to Campus Life.

- Make Reservations, currently the application only allows enroll in sports and cultural activities but does not allow reservations of tennis, tennis or golf game.
- Receive notifications you want the application to receive customized to the registered user activities and personal tastes and offers of interest can be notifications.
- Add search path on maps, you want to add functionality to find the desired route to the office marked on the map.
- Add a section with information about sports centers, facilities, etc. To enhance the feeling of better understanding the sports center that the customer can try to see in advance the facilities and services.

- Improved RRSS is intended to add more social networks and improve service sharing networks, for better interaction between applications of this kind.
- Download registration receipts and slips and purchase cards.
- View real-time waiting lists, to see real-time applications of an activity and to know how many spaces are free and how many need to be discarded to enroll in such activity.

These are some of the improvements you want to apply in future versions of the application, some materialize in the next version, some materialize in later versions, some of today could disappear and others were to ever implemented, everything depends on the progress of the application acceptance, use and needs, what is clear is that new versions will have to be improved for the customer is more satisfied every day.